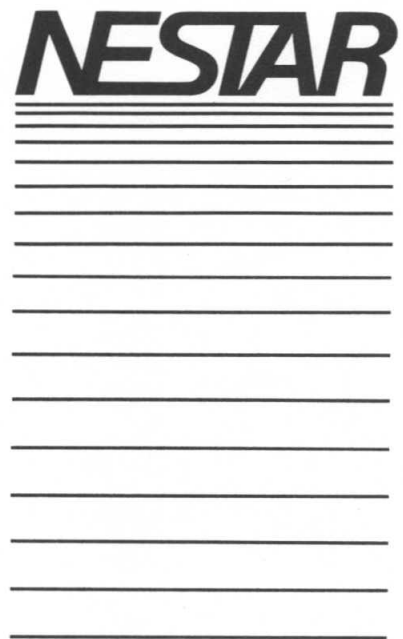


Cluster/One, Model A User Manual



SECRET

NESTAR SYSTEMS, INCORPORATED
The information in this document is the property of Nestar Systems, Inc. and is to be controlled, stored, and handled in accordance with the policies and procedures of Nestar Systems, Inc. This document is to be controlled, stored, and handled in accordance with the policies and procedures of Nestar Systems, Inc.

NESTAR SYSTEMS, INCORPORATED

CLUSTER/ONE MODEL A (TM) Version 1.0

USER'S MANUAL

**** FIRST EDITION ****

Release Date: July 2, 1980

This document is the property of Nestar Systems, Inc. and is to be controlled, stored, and handled in accordance with the policies and procedures of Nestar Systems, Inc. This document is to be controlled, stored, and handled in accordance with the policies and procedures of Nestar Systems, Inc.

Copyright © 1980 by Nestar Systems, Inc.

1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890

1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890

1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890 1234567890

DISCLAIMER

NOTICE

Nestar Systems, Incorporated, reserves the right to make changes to any products described herein to improve their functioning or design.

DISCLAIMER OF ALL WARRANTIES AND LIABILITY

The information in this document has been carefully reviewed and is believed to be reliable; nevertheless, Nestar Systems, Incorporated makes no warranties, either express or implied, with respect to this manual or with respect to the software described in this manual, its quality, performance, merchantability, or fitness for any particular purpose. Nestar Systems, Incorporated software is sold or licensed "as is." The entire risk as to its quality and performance is with the buyer. Should the programs prove defective following their purchase, the buyer (and not Nestar Systems, Incorporated, its distributor, or its retailer) assumes the entire cost of all necessary servicing, repair, or correction and any incidental or consequential damages. In no event will Nestar Systems, Incorporated be liable for direct, indirect, incidental, or consequential damages resulting from any defect in the software, even if Nestar Systems, Incorporated has been advised of the possibility of such damages. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you.

This document is copyrighted and contains proprietary information. All rights are reserved. This document may not, in whole or part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from Nestar Systems, Incorporated.

Cluster/One Model A (TM)
Copyright 1980 by Nestar Systems, Incorporated
430 Sherman Avenue, Palo Alto, California 94306
(415)327-0125

Part number 2001-0100-0
All rights reserved.

How to Use This Manual

This is Release One of the Cluster/One Model A User's Manual. It is designed for experienced programmers who know the DOS, BASIC, and Pascal programming languages. This manual is designed as both a tutorial for new users and a reference manual for experienced users. New users should see especially:

Chapter 1, an overview

Chapter 2, a hands-on tutorial for a bare-bones system

There are also chapters on disk organization, protection, and using Cluster/One with DOS and Pascal.

For reference purposes, users should see:

Chapter 5, where individual commands are described in alphabetical order

Appendix E, a quick reference list of commands and parameters

Appendix A, a list of error and information messages

We welcome criticisms and suggestions. Forms for reporting program bugs and documentation inadequacies are included near the back of the manual.

Table of Contents

How to Use This Manual.....	ii
Chapter 1 Introduction.....	1-1
1.1 Up to 65 Apples Share a Disk System.....	1-1
1.2 DOS and Pascal Think They're Alone.....	1-1
1.3 Cluster Commands Control Virtual Diskettes.....	1-2
1.4 Central Disk System: Floppy Plus Optional Hard...1-2	
1.5 Dedicated User-Stations Control External Devices.1-3	
1.6 User-Stations and the Disk-Server.....	1-4
Chapter 2 The Bare Bones System.....	2-1
2.1 Virtual Diskettes.....	2-1
2.2 Virtual Drives.....	2-2
2.3 The Simplest Disk Organization.....	2-2
2.4 Registration and Logon Procedures.....	2-3
2.5 Issuing Direct Cluster Commands Under DOS.....	2-4
2.6 Initializing a DOS Virtual Diskette.....	2-8
2.7 Issuing Cluster Commands Within a BASIC Program..2-10	
2.8 Copying a Program Library.....	2-13
2.8.1 Copying from a Cassette Tape.....	2-13
2.8.2 Copying from a Mini-Diskette.....	2-13
2.9 Other Features.....	2-14
Chapter 3 Disk Organization and Pathnaming.....	3-1
3.1 Disk Organization.....	3-1
3.1.1 Disk Units.....	3-1
3.1.2 Cluster Files.....	3-1
3.1.3 Directories.....	3-2
3.1.4 Virtual Diskettes.....	3-2
3.1.5 Cluster Utilities.....	3-2
3.2 Pathnaming Conventions.....	3-3
3.2.1 Pathname: /unitnumber.....	3-3
3.2.2 Pathname: /unitname.....	3-3
3.2.3 Pathname: /unitname/subdirectory.....	3-3
3.2.4 Pathname: /unitname/subdirectory1 /subdirectory2/virtual diskette.....	3-4
3.2.5 Current Default Directory.....	3-4
3.2.6 Passwords.....	3-5
3.2.7 Pathnames--A Formal Definition.....	3-5

CONTENTS

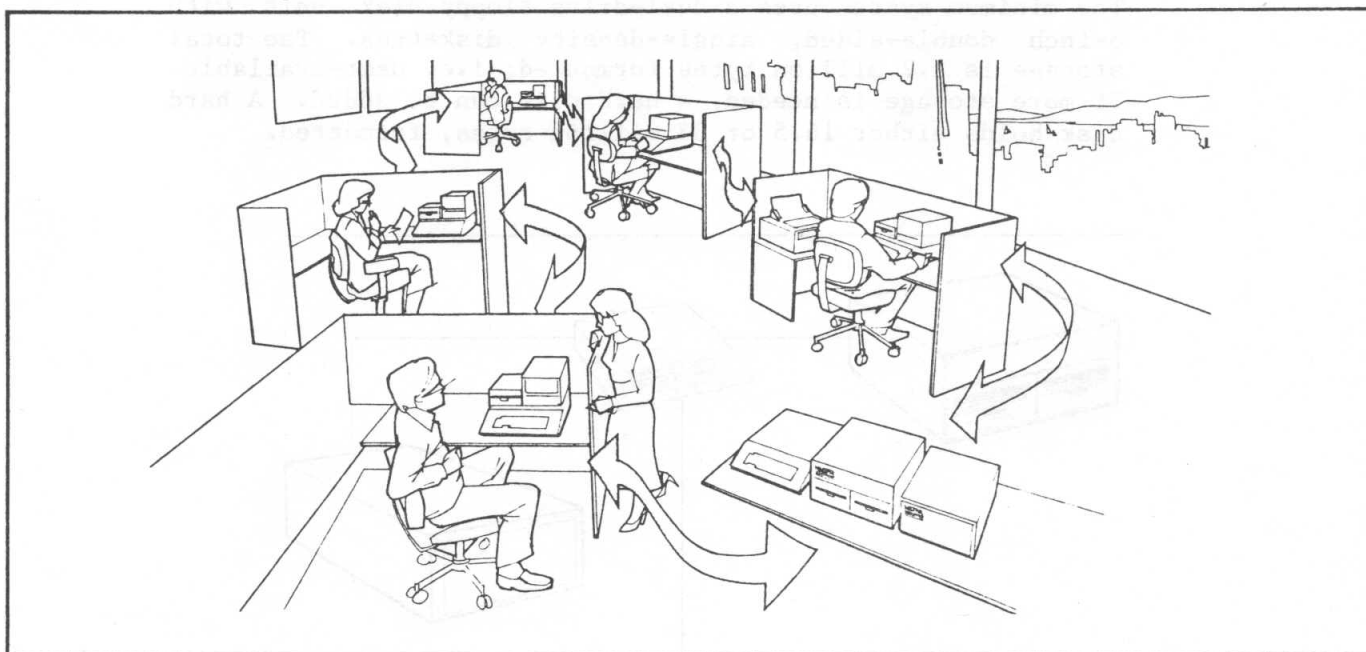
Chapter 4	Protection.....	4-1
4.1	Cluster-File Protection.....	4-1
4.1.1	User Classes.....	4-2
4.1.2	Passwords and Access Rights.....	4-3
4.1.3	Default Access Rights.....	4-4
4.1.4	The PROTECT Command.....	4-5
4.1.5	The SHOW PROTECT Command.....	4-6
4.1.6	The MOUNT Command.....	4-6
4.1.7	Preset Passwords.....	4-6
4.1.8	Access Rights Needed for Cluster Commands.....	4-7
4.1.9	What Happens If You Forget a Password.....	4-7
4.2	Locks.....	4-8
4.2.1	LOCK Command.....	4-8
4.2.2	UNLOCK Command.....	4-9
4.2.3	SHOW LOCK(S) Command.....	4-9
Chapter 5	Cluster Commands and Parameters.....	5-1
BLOAD.....		5-5
BRUN.....		5-5
BSAVE.....		5-5
CREATE.....		5-9
DELETE.....		5-13
HELP.....		5-14
LIST.....		5-15
LOCK.....		5-18
MOUNT.....		5-20
OFF.....		5-22
PATHNAME.....		5-23
PROTECT.....		5-24
PROTECTION.....		5-26
SET DIR.....		5-27
SET password.....		5-29
SHOW DATE.....		5-31
SHOW DIR.....		5-32
SHOW LOCK.....		5-33
SHOW LOCKS.....		5-34
SHOW MOUNTS.....		5-35
SHOW PROTECTION.....		5-37
SHOW STATION.....		5-38
SHOW TYPE.....		5-39
SHOW VOLS.....		5-40
UNLOCK.....		5-41
UNMOUNT.....		5-42

Chapter 6	Using DOS with the Cluster/One System.....	6-1
6.1	Size and Number of Sectors.....	6-1
6.2	Drives.....	6-1
6.3	Using Immediate Execution Mode in DOS.....	6-2
6.4	Using Cluster Commands within a BASIC Program....	6-2
6.4.1	Applesoft BASIC.....	6-3
6.4.2	Integer BASIC.....	6-4
6.5	Initializing a Virtual Diskette in DOS.....	6-5
Chapter 7	Using Pascal with the Cluster/One System...	7-1
7.1	Virtual Diskettes of Variable Size.....	7-1
7.2	Drives.....	7-2
7.3	CMD.....	7-3
7.3.1	SHOW DRIVES.....	7-4
7.3.2	SET D<drive>,drive-status.....	7-4
7.3.3	SHOW SLOT.....	7-5
7.3.4	SET S<slot>.....	7-5
7.3.5	SET D<drive>,S<slot>.....	7-5
7.4	Using Cluster Commands Within a Pascal Program...	7-5
7.5	Initializing a Virtual Diskette in Pascal.....	7-8
Appendices		
A	Error and Information Messages.....	A-1
B	The BOOT Program.....	B-1
B.1	Description.....	B-1
B.2	Operation of BOOT.....	B-1
B.2.1	Booting DOS Virtual Diskettes.....	B-2
B.2.2	Booting Pascal Virtual Diskettes.....	B-3
C	NESTAR DISK COPY Utility Program.....	C-1
C.1	Description.....	C-1
C.2	General Operation.....	C-1
C.3	Specific Operation.....	C-2
C.4	Technical Notes.....	C-3
D	Reading the Station Number from a Program.....	D-1
D.1	Introduction.....	D-1
D.2	Machine Language.....	D-1
D.3	(Applesoft) BASIC.....	D-1
D.4	Pascal.....	D-2
E	Quick Reference List - Cluster Commands.....	E-1

1. Introduction

1.1 Up to 65 Apples Share a Disk System

The Cluster/One Model A system, developed by Nestar Systems, Incorporated of Palo Alto, California, is a hard-wired network of up to 65 Apple microcomputers which share a disk storage system. One Apple acts as a disk-server, transferring programs and data between the disk storage and the Apples which act as user-stations. Except during such disk transfers, the individual user-stations function as independent computers with full storage and computing capabilities.



1.2 DOS and Pascal Think They're Alone

To the individual user, the central disk storage system is a set of virtual mini-disks which are indistinguishable from the 5 1/4 inch floppies on a stand-alone Apple. Almost all programs which run on a stand-alone Apple can run without modification on the Cluster/One. Apples which have appropriate hardware (e.g. a language card) can run under either the Pascal Operating System or the Disk Operating System (DOS) for integer and Applesoft BASIC. A Pascal program can have as many as six virtual mini-disks mounted (i.e., accessible by the program) at the same time, and a BASIC program can have as many as 255.

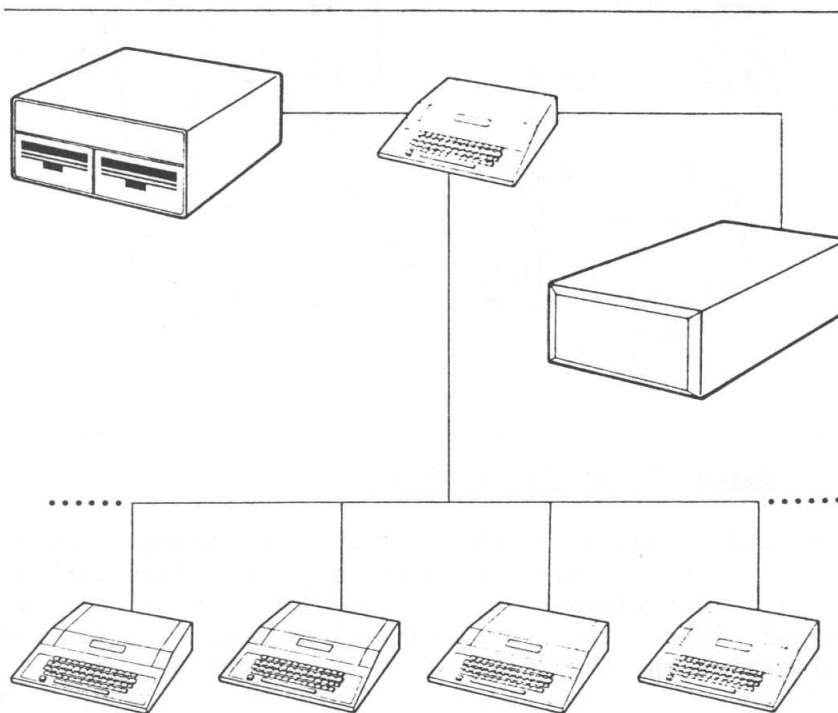
INTRO

1.3 Cluster Commands Control Virtual Diskettes

Cluster commands are available for creating, mounting, and protecting virtual mini-diskettes and for displaying the status of the Cluster directories, mounts, and locks. Once a virtual mini-diskette has been mounted, it is available to either DOS or Pascal for the full range of file operations: catalogues and directories can be listed; programs can be loaded and saved; and data files can be read and written, just as with a real floppy.

1.4 Central Disk System: Floppy Plus Optional Hard

The minimum system uses a dual-drive floppy disk unit with 8-inch double-sided, single-density diskettes. The total storage is 1.2 million bytes formatted; i.e. user-available. If more storage is needed, a hard disk can be added. A hard disk holds either 16.5 or 33 million bytes, formatted.

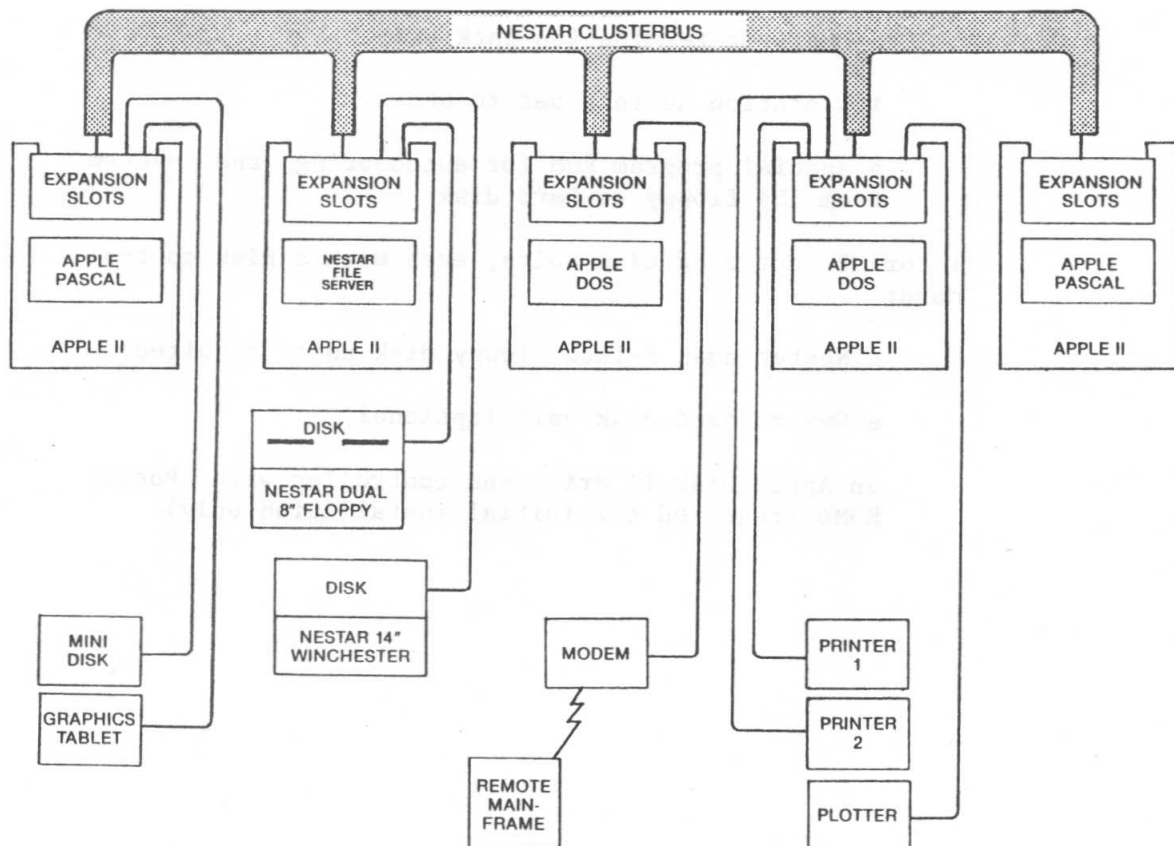


The Nestar "Model A" system uses one dedicated Apple as central mass storage manager which connects with all the user stations through the Nestar Cluster Bus.

1.5 Dedicated User-Stations Control External Devices

Each user-station can control external devices for its own private use. If an external device, say the printer, is to be used by more than one user, then one of the user-stations can listen for and execute requests for printer service. Though dedicated in the sense that it listens periodically for requests, the station can still perform other tasks as long as they do not interfere with its dedicated function. Examples of functions which could be serviced by dedicated stations and thus made available to more than one user are:

collecting data from a scientific instrument
 interfacing with a phone line to a remote user
 reading positions from a graphics tablet
 formatting and printing text for word processing
 functioning as a time-of-day clock
 plotting results.



1.6 User-Stations and the Disk-Server

The Apples which function as user-stations on the Cluster/One system must have a Nestar-supplied network interface card inserted in an expansion slot (other than 0). If the Apple is connected to more than one network, additional network card(s) are inserted in other slots. If the user station runs under the Pascal Operating System, it must have a 48K memory and a Language Card in slot 0. DOS requires only a 32K memory, or even 16K for some minimum applications. DOS requires a Language Card or ROM only if both Integer and Floating Point BASIC are to be used.

The Apple which functions as a disk-server must have:

- 48K RAM and a Language Card

- a special Nestar disk-server network card in slot 6 which differs from the user-station network card in the following ways:

 - a resistor-termination-pack chip

 - its station address set to \$FE

 - a special program ROM for autobooting the system from the floppy or hard disk

- 1 or 2 attached disk units, each with a disk control card:

 - a Nestar dual 8-inch floppy disk unit (required)

 - a Nestar hard disk unit (optional)

 - an Apple Disk II drive and controller with Pascal ROMs (required for initial installation only)

Chapter 2

The Bare Bones System

The Cluster/One Model A is a sophisticated system involving complex file-naming conventions and protection procedures. But in this chapter, which is designed as a hands-on tutorial for a beginning user, it will be treated as a bare-bones, no-frills system. This chapter will present the framework of the system. Once the fundamental concepts are mastered, the more sophisticated features presented later will be seen as logical elaborations.

Another simplification in this tutorial chapter, is that the examples will all be for DOS. The relationship of Cluster/One to DOS is similar but not identical to its relationship to the Pascal Operating System. However, once the rules for issuing Cluster commands under DOS are understood, the trivially different rules for Pascal will be easily understood. For more details on using Cluster commands within each operating system, see Chapter 6 for DOS and Chapter 7 for Pascal.

The fundamental concepts of the Cluster/One Model A system are:

- virtual diskettes
- virtual drives
- the simplest file structure
- addressing virtual diskettes
- getting started
- issuing direct Cluster commands under DOS
- the essential Cluster commands:

- SET DIR
- CREATE
- DELETE
- MOUNT
- UNMOUNT
- LIST
- SHOW MOUNTS

- initializing a virtual diskette under DOS
- issuing Cluster commands within a BASIC program.

2.1 Virtual Diskettes

Nestar's philosophy is to allow each individual user to program with exactly the same DOS, BASIC, or Pascal commands he uses with a stand-alone Apple, and, in addition, to use as many virtual mini-diskettes as he needs. With a stand-

alone Apple, a large data-base program requires constant insertion and removal of mini-diskettes in the 5 1/4 inch disk drives; Nestar provides the same capability by allowing the user to create, delete, mount, and unmount virtual mini-diskettes which reside on the central disk system. Each virtual diskette has the same internal structure, controlled by either the DOS or Pascal Operating System, that a real mini-diskette has. Since the formats required by DOS and Pascal are different, DOS and Pascal programs and data cannot be stored on the same virtual diskette.

Even though there can be as many as 64 users all sharing the central disk system, each individual user still operates as if he were using a stand-alone Apple with his own private mini-disk drives. The only difference is that instead of using a mini-floppy diskette, he issues a Cluster command to create a virtual diskette on the central disk system, and instead of physically inserting a real diskette into a real mini-disk drive, he issues a Cluster command to mount a virtual diskette on a virtual drive. DOS and Pascal don't know the difference.

2.2 Virtual Drives

A real disk drive must have a floppy diskette inserted before it is useful; correspondingly, a virtual drive must have a virtual diskette mounted before it is useful. The CREATE command allocates disk space which is equivalent to a mini-floppy diskette and is referred to in this manual as either a virtual diskette or a volume. The CREATE command also puts the name and address of the virtual diskette in a directory. The MOUNT command establishes a correspondence between a virtual diskette and a virtual mini-disk drive; it sets pointers so that when DOS is instructed to read or write on a real diskette inserted in physical drive n, it will read or write instead on the virtual diskette corresponding to (MOUNTED on) virtual drive n.

2.3 The Simplest Disk Organization

Each system floppy diskette or hard disk drive is divided into:

- Cluster directories
- virtual diskettes
- Cluster utility programs.

In the simplest case there is just one directory (called a root directory) on a given 8 inch system floppy diskette or hard disk. This directory contains the names and addresses of all virtual diskettes and Cluster utility programs stored

on that disk unit; it cannot contain names of virtual diskettes or Cluster utility programs stored on other disk units.

To address a given virtual diskette, the user specifies a pathname which contains:

the name or number of the physical disk unit the virtual diskette is stored on

the name of the virtual diskette itself.

For example, if a file is called LINDA and is stored on the diskette inserted in the left drive of the first system floppy (unit 1), the pathname is:

/1/LINDA

Obviously, the system floppy could just as easily be inserted in the right hand drive (unit 2) or, for faster access, copied onto a hard disk. Therefore, it is often more convenient to give the name of the physical unit rather than the number corresponding to its current location. For example, with each Cluster/One Model A system, Nestar supplies a system disk called MAIN. Included on MAIN is a virtual diskette called BOOT. To address BOOT, irrespective of where the system disk is physically located, the pathname is:

/MAIN/BOOT

In a slightly more complex case, each root directory can contain the names and addresses of subdirectories, as well as, or, in place of, the names of virtual diskettes and utility programs. For example, in a typical installation, a root directory may have an entry called USERS. USERS, in turn, is a list of subdirectories, one for each user--e.g. TOM, DICK, HARRY, MILHOUS.

2.4 Registration and Logon Procedures

The remainder of this chapter is intended as a hands-on tutorial to get you started with the Cluster/One Model A system. Since all installations are different, ask your system manager how to register and log on at a user terminal. For readers who do not have access to a user-terminal while reading this chapter, the procedure supplied by Nestar is described.

Ask your system manager to give you the name of a public DOS disk which you can use to try out Cluster Commands and create your own private virtual diskette. For purposes of

this tutorial, we shall assume that he gives you the name:

/MAIN/DOS/3.2.1

At a user-station, turn the Apple power switch (left, rear) off and on to ensure a fresh start. (Later you will learn to accomplish the same thing by typing @OFF when your Apple has been set to accept Cluster commands.) The screen will display:



CLUSTER/ONE MODEL A NETWORK BOOT 1.0

STATION ADDRESS nn

VOLUME TO BOOT: /MAIN/

where nn is the number of your station (in hexadecimal).

Enter the pathname exactly as your system manager gave it to you. You can use the backarrow to backspace to the M of MAIN if your virtual diskette is on some other disk unit. For the Nestar example, type DOS/3.2.1. The last line will now read:

VOLUME TO BOOT: /MAIN/DOS/3.2.1

(Note: Terminate all inputs to the Cluster/One system with a carriage return, except in special cases where the expected response is always a single character.)

If everything was entered correctly, the speaker will click softly for a few seconds to indicate network transactions and a display similar to the following will appear:

MASTER DISKETTE VERSION 3.2.1 PLUS

31-JULY-79

COPYRIGHT 1979 APPLE COMPUTER INC.

NESTAR DISK COPY 1.0

COPYRIGHT 1980, NESTAR SYSTEMS, INC.

Then a prompt appears:

```
> (for integer BASIC)
or
] (for Applesoft BASIC)
```

2.5 Issuing Direct Cluster Commands Under DOS

Before entering Nestar commands, it is necessary to issue a command which sends all output to the slot containing the network interface card:

PR#n

If the slot number is 6, as it usually is, the command is:

PR#6

Now it is possible to enter Cluster commands and have them executed immediately. The procedure is:

1. Type an @ sign immediately followed by the Cluster command you wish executed. Hit carriage return. (Commands typed without an @ sign are directed to DOS or BASIC just as they would have been without the Cluster.)

2. a. If the disk-server can execute the command, it does so, and responds

0,OK.

- b. If the disk-server cannot execute the command, it returns an error code number and message. For example:

1,ILLEGAL COMMAND
30,DRIVE REQUIRED

3. The prompt again appears on the CRT.

Note: The numbers which precede the messages are there to be used when input to programs (see Sections 2.7 and 6.4) or as a reference to a more complete description in Appendix A.

In the following sequence, the commands all deal with a virtual diskette named EXAMPLE. If you are following these commands on an Apple, you may prefer to substitute for EXAMPLE the name of a virtual diskette you need to create and use.

Now enter the examples:

@SET DIR /MAIN

This command will set your default directory to the root directory on the disk unit called MAIN. This makes it unnecessary for you to give the directory name each time you specify a pathname. (If your system manager gave you some other virtual diskette to work with, in place of /MAIN enter all of the pathname except the word following the final slash.)

```
@CREATE EXAMPLE,T=D,SECTORS=13
```

This command will allocate space for a virtual diskette called EXAMPLE on the same physical disk that contains your booted virtual diskette--in our example, the disk unit MAIN. The command also enters the name EXAMPLE and its address in your default directory--in our example, the root directory of MAIN. The field T=D (type= DOS) tells the disk-server that this virtual diskette will be used to store DOS programs and data. The field SECTORS=13 tells the disk-server that DOS 3.2.1, which writes 13 sectors per track, will be the operating system for this diskette.

It is not necessary to specify the physical disk unit and the name of your directory, because they were set with the SET DIR command you just issued. If the pathname (in this case EXAMPLE) does not begin with a slash, the system knows that it should use the current default directory.

If the command is entered correctly, the disk-server executes it and responds:

```
0,OK
```

You can now verify the CREATE by listing your directory. Enter:

```
@LIST
```

The system now lists on the screen all the virtual diskettes included in your directory. Among them will be the DOS directory and the EXAMPLE virtual diskette we just created.

```
/MAIN
  DOS,T=Y
  EXAMPLE,T=V
30635S FREE, 8 EXTENTS, 29764S LARGEST
```

The bottom line gives:

```
the total number of free sectors
on the disk unit
```

```
the number of extents--sets of
contiguous free sectors
```

```
the number of sectors in the largest
extent of free blocks
```

After each entry in the simple directory listing, the system lists the Cluster-file type:

T=V Virtual Diskette
 T=Y Directory
 T=S Cluster System Utility

To remove EXAMPLE from your directory, issue a DELETE command:

@DELETE EXAMPLE

The system should respond:

0,OK

and you can verify the DELETE by again entering:

@LIST

The virtual diskette EXAMPLE is now gone.

Re-enter the CREATE command:

@CREATE EXAMPLE,T=D,SECTORS=13

and after the

0,OK

enter a MOUNT command:

@MOUNT EXAMPLE,D2

(To DOS, this is equivalent to inserting a mini-floppy in drive 2 of a private disk drive attached to your stand-alone Apple. On the Cluster/One system, it involves setting pointers so that commands addressed to drive 2 will get and take information from the virtual diskette EXAMPLE.)

After the

0,OK

verify the MOUNT by displaying the current MOUNTS at your user-station:

@SHOW MOUNTS

Now on the screen will appear:

NAME	DRV	ACCESS	T	VOL	SIZE
/MAIN/EXAMPLE					
	2	RO SHR	D	0	455S
/MAIN/DOS/3.2.1					
	1	RO SHR	D	254	227S

The system automatically mounted the DOS diskette on drive 1 when you began the session on your user-station. It mounted EXAMPLE on drive 2 when you entered the MOUNT command. Because these examples are for a no-frills system, we are not entering any optional parameters. Therefore, by default, our virtual diskette EXAMPLE was mounted for shared usage (SHR) which means that other users are allowed to mount it at the same time. Also by default, EXAMPLE is mounted with read only access rights. A booted DOS virtual diskette, in this case DOS 3.2.1, is always mounted for read only shared usage (unless the BOOT program has been modified at your installation).

To undo the MOUNT (perform the equivalent of removing the diskette from real mini-drive 2) enter:

```
@UNMOUNT D2
```

This can be verified by again entering:

```
@SHOW MOUNTS
```

We have now tried out seven fundamental commands:

```
SET DIR
CREATE
DELETE
MOUNT
UNMOUNT
LIST
SHOW MOUNTS
```

The next step is to initialize the virtual diskette EXAMPLE, so that you can give its name when the system says VOLUME TO BOOT?

2.6 Initializing a DOS Virtual Diskette

As explained in the Apple publication "DOS Version 3.2", a mini-diskette must be initialized before it can be used to store BASIC programs and data. This is also true for virtual diskettes on the Cluster/One Model A. To initialize a virtual diskette, it must be mounted for exclusive use with read/write access rights.

Therefore, to initialize EXAMPLE we mount it with the following command:

```
@MOUNT EXAMPLE,EXC,RW,D1
```

Now, following the example on page 14 of "DOS Version 3.2", we write a greeting program, such as:

```
NEW
10 REM GREETING PROGRAM
20 PRINT "VIRTUAL DISKETTE EXAMPLE"
30 PRINT "CREATED today's date BY your name"
40 END
```

Run and debug the greeting program, then initialize the virtual diskette by entering:

```
INIT HELLO,V1,S6,D1
```

This DOS command formats the virtual diskette and saves the greeting program under the name HELLO so that it will be run whenever the virtual diskette is booted.

When the prompt appears again, return to the BOOT program by typing

```
@OFF
```

This time, respond to the prompt

```
VOLUME TO BOOT: /MAIN/
```

with the new pathname:

```
/MAIN/EXAMPLE
```

This time your new diskette will be mounted on drive 1 and the screen should display:

```
VIRTUAL DISKETTE EXAMPLE
CREATED today's date BY your name
```

Now you are ready to write, save, and run BASIC and DOS programs, just as you would on a stand-alone Apple. Take a few minutes to verify this with some simple DOS commands.

First, remount EXAMPLE with read/write access so that you can save programs on it:

```
PR#6
@MOUNT /MAIN/EXAMPLE,D1,RW
```


Then write a simple program, save it, display its name in your catalog, and run it. Your screen will look something like this:

```
NEW
10 PRINT "MY PROGRAM"
20 END
```

```
SAVE PROGRAM1,S6,D1
```

```
CATALOG
```

```
A 002 HELLO
A 002 PROGRAM1
```

```
RUN PROGRAM1,S6,D1
```

```
MY PROGRAM
```

2.7 Issuing Cluster Commands Within a BASIC Program

All of the Nestar commands have so far been typed directly at the keyboard. It is often convenient to issue the same commands from a program, and this is possible for all of the Cluster commands used in DOS. For example, at the beginning of each session you may want to set your default directory and mount your virtual diskette with read/write access so you can save programs. This can be accomplished by including the Cluster commands

```
SET DIR /MAIN
```

```
MOUNT EXAMPLE,D1,RW,EXC
```

in your greeting program. This is also a simple example of how to use Cluster commands within a BASIC program.

Integer and Applesoft BASIC use slightly different commands to communicate with the disk-server. The following example is for Applesoft only. If your Apple has only Integer BASIC, see Section 6.4.2.

Enter the following Applesoft program at your user-station:

```
10 REM GREETING PROGRAM USING CLUSTER COMMAND
20 HOME: VTAB3
30 PRINT CHR$(4);"PR#6"
```

CHR\$(4) is a CTRL-D. It signifies that the next field is a DOS command. "PR#6" enables the network card to look at all subsequent print

statements to see whether they contain Cluster commands.

```
40 PRINT "VIRTUAL DISKETTE EXAMPLE"
50 PRINT "CREATED today's date BY your name"
60 PRINT CHR$(14) "SET DIR /MAIN"
```

CHR\$(14) is a CTRL-N. It signifies that the next field is a Cluster command. Note that no @ sign precedes a Cluster command when it occurs within a program.

```
70 GOSUB 500
```

500 contains a subroutine to input and display the return code and message

```
80 PRINT CHR$(14); "MOUNT EXAMPLE,D1,RW,EXC"
90 GOSUB 500
100 PRINT CHR$(4);"CATALOG"
110 END
```

```
500 REM INPUT AND DISPLAY RETURN CODE AND MESSAGE
510 PRINT CHR$(4); "IN#6"
```

CHR\$(4) again signifies a DOS command. "IN#6" tells the following INPUT command to take input from slot 6 rather than from the keyboard.

```
520 INPUT NN,NN$
```

The file-server sends a return code (NN) and message (NN\$) to the network card in slot 6. After reading the last character from slot 6, the INPUT command resets the input slot so the next input will come from the keyboard.

```
530 PRINT NN;" ";NN$
540 RETURN
```

If the network interface card is in some slot other than 6, that slot number must be substituted in the PR#6 and IN#6 commands.

Run and debug the new greeting program. The MOUNT command within the program will mount the virtual diskette EXAMPLE with read/write access so that you can now save programs on it. Therefore, you can replace the previous HELLO program by issuing the DOS command:

```
SAVE HELLO,S6,D1
```

BARE BONES

Again reboot by typing
@OFF

This time when the system asks

VOLUME TO BOOT?

reply with the pathname:

/MAIN/EXAMPLE

Your screen should show:

VIRTUAL DISKETTE EXAMPLE
CREATED today's date BY your name

0,OK

0,OK

DISK VOLUME 1

A 002 HELLO
A 002 PROGRAM1

You now know how to:

CREATE a virtual diskette

MOUNT it on a virtual drive

Initialize it

Boot it

Issue commands and receive responses from
within programs

This means you are ready to use your Apple computer just as you do a stand-alone Apple. You can write programs and save them on your virtual diskette, then LOAD and RUN them, just as you have always done, but now using shared disks in a local network of Apples.

2.8 Copying a Program Library

If you already have a library of programs and data files for your Apple, either on cassette tapes or mini-diskettes, you can copy them onto your virtual diskette just as you would with a stand-alone Apple.

2.8.1 Copying From a Cassette Tape

1. Create and mount the virtual diskette on which you wish to copy files. (We assume the diskette will be mounted with read/write access on drive 1 and the network interface card is in slot 6.) For example:

```
MOUNT /MAIN/EXAMPLE,D1,RW,EXC
```

2. Connect the tape cassette to the special plug at the rear right of the Apple.

3. Give the DOS commands:

```
LOAD
```

```
SAVE filename,S6,D1
```

(See the DOS manual for more details on loading programs from a cassette.) Repeat the two DOS commands in step 3 until all your files are copied.

2.8.2 Copying from a Mini-Diskette

NOTE: Appendix C describes the Nestar Disk Copy program which copies whole diskettes (including data files). The following procedure is to be used when only selected programs are to be copied.

1. Create and mount the virtual diskette on which you wish to copy files. Be sure the number of sectors per track is the same for the real and virtual diskettes--i.e., they must both have been initialized under DOS 3.3 (16 sectors per track) or under an earlier version of DOS (13 sectors per track). (We assume that the network card is in slot 6 and that the virtual diskette is mounted with read/write access on drive 1.)

2. Connect the mini-disk to drive 1 of a disk control card in slot 5 (or any other available drive and slot).

3. Give the two DOS commands:

```
LOAD filename,S5,D1
```

SAVE filename,S6,D1

Repeat the DOS commands in step 3 until all the files are copied.

There is also a Nestar-supplied utility called NESTAR DISK COPY which simplifies the process of copying entire diskettes. This is described in Appendix C.

2.9 Other Features

For many applications what has been described so far is all you need to know. But Cluster/One also serves the user who needs to:

- organize his disk into root directories, subdirectories, and virtual diskettes of various types and sizes

- protect directories and virtual diskettes by requiring passwords and defining access rights for different classes of users

- allow several users to set locks in order to cooperate in sharing system resources

- use binary virtual diskettes

- use the Pascal Operating System.

These features are described in the remainder of this manual.

Chapter 3

Disk Organization and Pathnaming Conventions3.1 Disk Organization3.1.1 Disk units

The central disk system consists of some combination of 8-inch floppy and hard disk drives. Each disk-unit is referred to by either its assigned name or its unit number. Disk units may be given names when they are formatted by a Nestar-supplied utility program. Unit numbers are:

Unit #	Unit or Device
1	diskette in left drive of floppy disk
2	diskette in right drive of floppy disk
5	hard disk

3.1.2 Cluster Files

The file structure on each physical disk unit has two distinct levels:

1. the Cluster level, which is unknown to DOS and Pascal, and is equivalent to a set of cardboard boxes, each containing a number of real mini-floppy diskettes
2. the DOS and Pascal level, where program and data files are stored on a single mini-diskette.

The Cluster-level files on each disk unit consist of:

a root directory
subdirectories
virtual diskettes
Cluster utility programs.

Note: In this manual, the expression "Cluster-file" will refer to these four types of Cluster-level files. The word "file" alone will be reserved for the program and data files contained within a single virtual diskette and controlled by the DOS or Pascal operating systems. The expression "virtual diskette" is equivalent to "volume" as used by DOS and Pascal. The expressions "Nestar File Server", "Cluster

DISK

File Server", and "disk-server" are used interchangeably to refer to the Apple which executes Cluster commands and transfers information to and from the central disk system.

3.1.3 Directories

The root directory on a given physical disk unit points, either directly or through subdirectories, to every virtual diskette, utility program, or subdirectory on that unit. It cannot point to Cluster-files on other physical units.

Each root directory and subdirectory consists of names and addresses of subdirectories, virtual diskettes and/or Cluster utility programs on the same physical unit as the directory itself. The size of a directory is determined by the file system and cannot be specified by the user. A directory which contains the name and address of a given Cluster-file will be referred to as its parent directory.

3.1.4 Virtual Diskettes

Virtual diskettes are of three types:

- DOS
- Pascal
- Binary

DOS virtual diskettes are always the size of a real mini-diskette, but Pascal and binary diskettes can vary in size, as specified by the user. See Chapters 6 (DOS) and 7 (Pascal) for details.

3.1.5 Cluster Utilities

Cluster utility programs are used to:

- start (boot) the system
- copy and format system floppies and hard disks
- back-up the system
- test the hardware.

These are described in the System Manager's Manual.

3.2 Pathnaming Conventions

To address a Cluster-file on the Cluster/One system, it is necessary to specify, by some combination of defaults and explicit names:

- the physical disk unit (i.e its root directory)
- intermediate subdirectories (if any)
- the desired Cluster-file--a subdirectory, virtual diskette or Cluster utility.

To address the root directory itself, only the physical disk unit number or name needs to be specified.

NOTE: Four commands (SET DIR, PROTECT, SHOW PROTECT, and LIST) can have a null pathname, in which case they operate on the current default directory. Section 3.2.5 describes how to set and use the current default directory.

First of all, let's look at a few examples of how pathnames are used with the LIST and CREATE Cluster commands:

3.2.1 Pathname: /unitnumber

LIST /1

This command lists all the Cluster-files whose names and addresses are included in the root directory on disk unit 1 (the 8-inch floppy diskette inserted in the left-hand drive of the first system floppy disk). The initial slash indicates the first field is the name or number of a physical disk unit.

3.2.2 Pathname: /unitname

LIST /MAIN

This command lists all the Cluster-files whose names and addresses are included in the root directory of the system floppy diskette or hard disk called MAIN, regardless of where it is physically located.

3.2.3 Pathname: /unitname/subdirectory

LIST /LANGUAGES/FRENCH

This command searches for the physical disk unit named LANGUAGES, and looks in its root directory for the name and address of the subdirectory FRENCH. It then reads FRENCH and displays all the Cluster files listed in it.

DISK

3.2.4 Pathname: /unitname/subdirectory1/subdirectory2/virtual diskette

```
CREATE /MAIN/USERS/JANE.DIR/JANE.V1,T=D,SECTORS=16
```

This command:

searches for the physical disk unit MAIN

looks in its root directory for the subdirectory
USERS

looks in USERS for JANE.DIR

allocates space on the physical disk unit MAIN
for the DOS-type virtual diskette JANE.V1

enters the name and address of JANE.V1 in the
subdirectory JANE.DIR.

If the subdirectories USERS and JANE.DIR in the above example do not exist, the CREATE command creates them as needed.

3.2.5 Current Default Directory

In order to cut down on the number of disk accesses and the number of characters to be typed and processed, the user may set a current default directory. Whenever a pathname is specified which does not begin with a slash, it is prefixed by the current default directory. In the previous example, assume that the subdirectories USERS and JANE.DIR exist before the CREATE command is issued. If so, Jane will achieve the same results if she enters:

```
SET DIR /MAIN/USERS/JANE.DIR  
CREATE JANE.V1,T=D,SECTORS=16
```

The SET DIR command above sets the current default directory to:

```
/MAIN/USERS/JANE.DIR
```

which is prefixed to the JANE.V1 when the CREATE command is executed. Thus the above two commands are equivalent to:

```
CREATE /MAIN/USERS/JANE.DIR/JANE.V1,T=D,SECTORS=16
```

The LIST, PROTECT, and SHOW PROTECT commands can be specified with no pathname; if so, each operates on the current default directory. A SET DIR with no pathname

clears the current default directory. The SHOW DIR command displays the name of the current default directory.

3.2.6 Passwords

As will be explained more fully in Chapter 4 (Protection), it is possible when creating a Cluster-file to require that a password be specified each time the Cluster-file is accessed for some or all operations. In our last example, let's suppose the passwords needed are:

```
COFFEE (for the root directory MAIN)
TEA    (for the subdirectory USERS)
MILK   (for the subdirectory JANE.DIR)
```

If her current default directory is set for some other directory, Jane would enter:

```
CREATE /MAIN:COFFEE/USERS:TEA/JANE.DIR:MILK/JANE.V1,T=D,
      SECTORS=16
```

Or alternatively, she can enter:

```
SET DIR /MAIN:COFFEE/USERS:TEA/JANE.DIR:MILK
CREATE JANE.V1,T=D,SECTORS=16
```

There are also commands to set default passwords so they do not have to be entered each time the pathname is specified; these will be explained in Chapter 4.

3.2.7 Pathnames--A Formal Definition

In general terms, a pathname is specified according to the following definitions. Square brackets indicate optional parameters, e.g. [:password].

```
pathname: /unitname/filepath
          or filepath
```

```
unitname: 1, 2, or 5 [:password]
          or name [:password]
```

```
filepath: name [:password] [/filepath]
```

```
name:      the name of a disk unit (root
          directory), subdirectory, virtual
          diskette or Cluster utility. A name
          has the following restrictions:
```

1-15 characters.

No commas, slashes, colons,

DISK

carriage returns, or unprintable characters.

Imbedded blanks are okay.
Leading and trailing blanks are removed by the system.

The first character can be any acceptable character.

password: a secret word used to restrict access to a directory or virtual file. Same restrictions as name.

Note that this is a "recursive" definition in that a form for filepath includes a filepath. Another way to describe a legal filepath is to say that it consists of one or more names separated by slashes, and each name may optionally be followed by a colon and a parameter. Recursive definitions will be used elsewhere in the manual to concisely describe sequences of terms.

Three other restrictions apply to the total pathname:

1. The total pathname cannot exceed 80 characters.
2. Upper or lowercase can be used for any characters within a Cluster command, and are equivalent.
3. An initial slash indicates the pathname begins in the root directory of a physical disk unit. If there is no initial slash, the current default directory is prefixed to the pathname.

Chapter 4

Protection

The Cluster/One allows many users to share a large data base, something that is barely feasible for stand-alone Apples. In a classroom, students can share a large selection of courses; in a factory, many different user-stations can update a central inventory; in a business office, the billing and advertising departments can share an address list. However, universal access is not always desirable: files must not be accessed at sensitive times or by unauthorized users. A payroll application must foil the potential computer crook; a classroom manager must make sure the students don't change their grades or copy the answers to a final exam; and inventory programs must make sure two users are not updating the inventory at the same time.

The Cluster/One Model A system has two different software mechanisms for protecting the shared resources of the system:

1. Cluster-file protection, a system of passwords and access rights for different classes of users
2. locks, which allow users to share any resource; for example, external devices or individual files on a virtual diskette.

These two protection systems differ in the following ways:

Locks can be used to control access to any resource--e.g., a printer, plotter, scientific instrument, or individual file on a virtual diskette. Passwords and access rights control access only to whole virtual diskettes and whole Cluster directories.

Locks are voluntary; they merely provide a convention that cooperating stations can use. The system returns the information that a lock is in use or in exclusive use, but it does not prevent the user from addressing the printer, the particular file, or whatever he is trying to protect. In contrast, passwords and access rights, once set, are involuntary; the system will not access a given virtual diskette or directory unless the protection information stored with it specifically allows it.

4.1 Cluster-File Protection

Each Cluster-file can have three different classes of users

PROTECTION

who have different passwords and are allowed to do different things. When someone creates a Cluster-file, he specifies for each class of user the password required and the access rights allowed.

This section describes:

- the user classes and access rights

- how to assign passwords and access rights with the CREATE and PROTECT commands

- what the default access rights are for each class of user

- how to MOUNT a virtual diskette with read only or read/write access right

- preset passwords

- access rights needed to execute various Cluster commands

4.1.1 User Classes

Potential users of a given Cluster-file are divided into three classes:

- public (PUB) users who do not know a password

- group (GRP) users who know the group password

- private (PRV) users who know the private password

The access rights which may be granted or denied to each class of user are:

- R the ability to read the data part of the Cluster-file

- W the ability to write on the Cluster-file

- E the ability to erase the Cluster-file

- C the ability to create a new entry in the directory (meaningless for virtual diskettes)

- D the ability to delete an entry from the directory (meaningless for virtual diskettes)

For example, suppose a given Cluster/One installation wants

to have accounting programs on a virtual diskette called ACC2. They want ACC2 to be listed in a directory called ACCOUNTING which in turn will be listed in the root directory of the system hard disk--unit 5.

They want to limit access to the directory and virtual diskette as follows:

1. public: the public will have no access rights.
2. group a select group of employees will be able to:

R read the virtual diskette and
execute the programs on it

The group password will be SILVER.

3. private The person in charge of the programs will be allowed to:

R read the virtual diskette
W write on the virtual diskette
E erase the entire diskette

He will also be allowed to:

D delete entries from the directory
C create new entries in the directory

The private password will be GOLD.

4.1.2 Passwords and Access Rights

Passwords and access rights can be specified for a new Cluster-file by a CREATE command and for an existing Cluster-file by a PROTECT command. With [optional] parameters indicated by square brackets, the syntax of these two commands is:

CREATE pathname, type [,size] [,protection] [,REPLACE]

PROTECT pathname [,protection]

(For a complete discussion of CREATE and PROTECT see the individual command descriptions in Chapter 5.)

The optional protection parameter is the same for both CREATE and PROTECT and is defined as follows:

PROTECTION

protection: protect item [,protection]

protect items:

GRPPW=password (group password)

PRVPW=password (private password)

PUBACC=access rights (public access rights)

GRPACC=access rights (group access rights)

PRVACC=access rights (private access rights)

access rights: a subset (in any order) of the letters RWECD (read, write, erase, create, and delete, as described under "private" in above example)

To specify no access rights, the equal sign is immediately followed by the delimiter--comma, blank or carriage return.
For example:

PUBACC=,GRPACC=

For the above example, the CREATE command would be:

CREATE /5/ACCOUNTING/ACC2,T=P,PRVPW=GOLD,PRVACC=RWE,
GRPPW=SILVER,GRPACC=R,PUBACC=

This command creates a Pascal-type virtual diskette named ACC2 which is listed in the subdirectory ACCOUNTING which in turn is listed in the root directory of the hard disk. Users giving the password GOLD can read, write or erase ACC2, but users giving the password SILVER can only read it. Users who do not give the password GOLD or SILVER have no access rights.

This command also implicitly creates the subdirectory ACCOUNTING (assuming it does not already exist). Passwords and access rights for ACCOUNTING are the same as for ACC2, except that create and delete access rights are added to the private rights for the directory.

4.1.3 Default Access Rights

If public, group, or private access rights are not specified in a CREATE command, the Cluster/One system sets standard default values:

PUBACC=R (read)

GRPACC=PUBACC

PRVACC varies:

For a virtual diskette: RWE (read, write, erase)

For an explicitly created directory: RWECD
(read, write, erase, create, delete)

For an implicitly created directory: the
private access rights for the Cluster/file
being explicitly created plus CD--create
and delete.

Therefore, in the previous example, the field defining
private access rights is not necessary because they are
the same as the defaults.

If a Cluster-file has no private password, then all of
its private access rights are granted to any user,
whether he gives no password or some extraneous
password.

When a Cluster-file is created, all directories
specified in the pathname which do not already exist
are also created; these implicitly created directories
are given CD (create and delete) private access rights
plus the access rights specified for the Cluster-file
being explicitly created; they are also given the same
passwords.

4.1.4 The PROTECT Command

Usually access rights and passwords are set when a virtual
diskette or directory is first CREATED. The PROTECT command
allows the user to modify the access rights and passwords
previously set. The syntax is given in Section 4.1.2 and
Chapter 5. Protect items which are not specified by the
PROTECT command remain the same. Anyone giving the private
password for the specified Cluster-file can issue a PROTECT
command; access rights do not apply. If there is no private
password then anyone can issue the PROTECT command.

PROTECTION

4.1.5 The SHOW PROTECT Command

The SHOW PROTECT command displays the private, group and public access rights currently assigned to the specified Cluster-file. If no pathname is given, the access rights for the current default directory are displayed. Passwords are never displayed.

4.1.6 The MOUNT Command

In addition to the restricted rights which are controlled by the protection attributes of a virtual diskette, access rights may be (voluntarily) limited even further when the virtual diskette is actually used. The MOUNT command allows a user to mount a specified virtual diskette with either read-only or read-write access, provided that these are access rights his password entitles him to. For example, if a user gives the private password and has full access rights, he may want to mount his virtual diskette with read only rights so as to prevent accidental writes. Usage is an optional parameter for the MOUNT command:

[,usage]

where usage can be RO (read only) or RW (read/write) and SHR (shared) or EXC (exclusive). If RO or RW is not specified, the default of read only is used. If SHR or EXC is not given, the default is SHR for RO or not given, and EXC for RW. When booted, DOS virtual diskettes are mounted RO SHR, but Pascal virtual diskettes are mounted RW EXC.

4.1.7 Preset passwords

Two commands allow a user to preset his group and private passwords so he doesn't need to enter them each time he specifies a pathname:

```
SET GRPPW=password
SET PRVPW=password
```

These commands make it less likely that an unauthorized person will learn a password and also cut down on the amount of typing to be done when entering a pathname.

If passwords are in a program, others can read the listing and know what they are. For example, if a password is WOW and your program contains the two Cluster commands:

```
MOUNT /MAIN/GRADES:WOW,D9
MOUNT /MAIN/ANSWERS:WOW,D10
```

an enterprising student can read your program and learn the

password.

Instead, the following Applesoft BASIC program segment, shows how passwords can be used in a program without appearing in the listing:

```
PRINT "ENTER PASSWORD";
INPUT PW$
PRINT CHR$(14);"SET PRVPW=";PW$
PRINT CHR$(14);"MOUNT /MAIN/GRADES,D9
PRINT CHR$(14);"MOUNT /MAIN/ANSWERS,D10
```

4.1.8 Access Rights Needed for Cluster Commands

In order to execute a Cluster command, a user must have the READ access right for every directory in the pathname. He needs additional access rights for the following commands:

<u>Command</u>	<u>Access Rights Required</u>
CREATE	C (create) for parent directory
DELETE	E (erase) for virtual diskette or directory being erased D (delete) for parent directory
PROTECT	Must have private password for Cluster-File. Access rights irrelevant.
MOUNT	R (read) for virtual diskette W (write) for virtual diskette if RW specified.
BRUN	
BLOAD	R (read) for virtual diskette
LIST	or directory addressed by
SHOW PROTECTION	pathname
SHOW DIR	
SHOW TYPE	
BSAVE	W (write) for virtual diskette addressed by pathname

4.1.9 What Happens If You Forget a Password...

Passwords are actually stored in an encrypted form from which it is impossible to reconstruct the original name. If you forget a password, there is no way to discover it by dumping the disk. The only recourse is to have the System Manager use superuser rights to remove or change the

PROTECTION

password.

4.2 Locks

LOCK commands allow several programs to cooperate in sharing a resource on the Cluster/One system, such as a particular file, printer, or some other external device. Sharing a printer is an obvious application, since it would be less than useful for output from two programs to be printed at the same time.

Another example is using the Cluster/One for inventory control. If two users update the inventory at the same time, one of the transactions may be lost. Suppose a computer store has eight Apples on hand; employee A sells two Apples; employee B sells 3 Apples; and both employees enter their transactions at the same time.

For the first transaction, the computer reads the file, computes $8-2=6$, and sends the result (6) back to the file. For the second transaction, the computer reads the file before the 8 is changed to a 6, computes $8-3=5$, and sends its result (5) to the file. So the computer now erroneously thinks it has 5 instead of 3 Apples on hand.

4.2.1 LOCK Command

To avoid such errors, each program, before updating the inventory, can issue a lock command:

```
LOCK lockname [,usage]
```

where lockname is any name 1-15 characters in length (except ALL) which has been chosen to represent the resource being shared. Usage is either:

```
SHR  shared--more than one user can have  
      the lock set at a given time
```

or

```
EXC  exclusive--only one user can have  
      the lock set at a given time.
```

The system then returns one of the following messages:

```
0,OK
```

```
41,IN USE
```

```
42,IN EXC USE
```

```
(or some syntax error messages)
```

The IN USE message is returned if the lock command specifies EXC (exclusive) usage when another user has already set the specified lock. The IN EXC USE message is returned if the lock command specifies SHR (shared) usage when another user has set the specified lock for exclusive use.

If the program receives the 0,OK message it can proceed to use the printer or update the file or whatever, but if it receives a 41,IN USE or 42,IN EXC USE message it should either go into a wait loop or go on to some other task before reissuing the LOCK command.

4.2.2 UNLOCK Command

To remove a lock, one of the following commands is issued:

UNLOCK lockname

-releases lockname for use by LOCK
commands from other stations

or

UNLOCK ALL

-releases all locks held by this station

The locks held by a station are automatically unlocked when the station is booted. In particular, turning the station off and on or issuing the OFF command will release all locks.

4.2.3 SHOW LOCK(S) Command

A user can display locks in several ways:

SHOW LOCKS

-displays all locks currently set at his
own user station

SHOW LOCK lockname

-displays all user stations currently
using the specified lock

The following information was obtained from the records of the Department of the Interior, Bureau of Land Management, regarding the land owned by the United States in the State of California.

The total area of land owned by the United States in the State of California is approximately 100,000,000 acres. This land is owned by the United States in several different capacities, including as a trustee for the people of the State, as a landowner, and as a lessor.

LAND OWNED BY THE UNITED STATES

The following is a list of the various types of land owned by the United States in the State of California:

Public Lands

Public lands are those lands owned by the United States which are not reserved for any special purpose, and which are available for disposal to the public.

Reserved Lands

Reserved lands are those lands owned by the United States which are reserved for a special purpose, and which are not available for disposal to the public.

The following is a list of the various types of reserved lands owned by the United States in the State of California:

LANDS RESERVED FOR THE PEOPLE

The following is a list of the various types of lands reserved for the people of the State of California:

State Lands

State lands are those lands owned by the State of California which are reserved for a special purpose, and which are not available for disposal to the public.

County Lands

County lands are those lands owned by the County of California which are reserved for a special purpose, and which are not available for disposal to the public.

Chapter 5

Cluster Commands

A Cluster command consists of a command word followed by a variable number of parameters, some required and others optional. A pathname or subcommand word, if present, must immediately follow the command word; other parameters may occur in any order. The delimiter following a command word must be a blank and the delimiter following a pathname must be a comma; elsewhere, either a comma or blank may be used as delimiter. For simplicity, however, a comma will always be used in this presentation.

Any alphabetic character in a Cluster command can be either upper or lower case; they have the same meaning. For standard Apples, this is not useful, but for Apples with terminals which display upper and lower case, and from Pascal, it is convenient.

Remember that Cluster commands may have to be preceded by special characters, depending on the language environment. Briefly, these characters are:

@	for the BASIC command level
CHR\$(14)	for the BASIC program level
nothing	for the Pascal CMD program
nothing	for the Pascal SENDMSG function

See Chapter 6 (DOS) or 7 (Pascal) for further details.

Syntactically, a Cluster command has the following form:

```
<0-n blanks><command word><1-n blanks><subcommand word>
<1-n blanks><pathname><,>other fields>
```

A pathname may have imbedded blanks, so a comma must be used to delimit it from the following fields. No blanks may be imbedded in a

command word
subcommand word
other fields

See Section 3.2 for a full description of pathnames.

COMMANDS

A field may have one of three forms:

Keyword (e.g. REPLACE or ALL)

Keyword=Value (e.g. T=D)

Letter value (e.g. D9 for drive 9)

Optional parameters are indicated by square brackets, e.g. [optional]. The system assigns default values, where meaningful, when optional parameters are not specified; these are given in the descriptions of the individual commands.

Some of the parameters are explained more fully elsewhere in this manual:

pathnames	Section 3.2
protection	Section 4
size	
DOS	Section 6.1
Pascal	Section 7.1
drive number	
DOS	Section 6.2
Pascal	Section 7.2

In addition, a one-page summary of PATHNAMES and PROTECTION are filed alphabetically with the commands in this chapter.

A quick reference list of all available Cluster commands and parameters occurs at the end of this book. The commands and parameters are alphabetized, and a page reference to detailed descriptions follows each item.

Cluster commands can be divided into these classes:

1. Commands concerned with creating, deleting, naming and protecting virtual diskettes:

CREATE
DELETE
LIST
PROTECT
SHOW PROTECTION

2. Commands concerned with mounting virtual diskettes
at a user station:

a. For both DOS and Pascal

MOUNT
UNMOUNT
SHOW MOUNTS

b. For Pascal only. Executed within
CMD and therefore not executable
within a user's own program:

SHOW DRIVES
SHOW SLOT
SET S<slot>
SET D<drive>,S<slot>
SET D<drive>,drive-status

3. Commands concerned with binary Cluster-files

BLOAD
BRUN
BSAVE

4. Commands concerned with locks

LOCK
UNLOCK
SHOW LOCK lockname
SHOW LOCKS

5. Commands which set default values

SET DIR
SET GRPPW
SET PRVPW

6. Commands which gives information at terminal

HELP
?
SHOW DATE
SHOW DIR
SHOW STATION
SHOW VOLS

COMMANDS

The remainder of this chapter consists of detailed descriptions of each Cluster command. The commands are presented in alphabetical order, and each new command or group of commands starts on a new page. For each command, the following information is presented:

- the command syntax (e.g. DELETE pathname)

- an abbreviated statement of what the command does

- parameters and defaults

- error and information messages

- access rights required to execute the command

- discussion

- examples

All error and information messages are listed in Appendix A. Individual command descriptions include only those messages which are specific to the given command.

BSAVE pathname, FROM=number, size, [,LOAD=number] [,REPLACE]

- saves memory image of specified size
starting at FROM address on
specified binary virtual diskette

BLOAD pathname [,LOAD=number]

- loads binary virtual diskette previously saved
by BSAVE
- loads at LOAD address in BLOAD, if present;
if not, loads at LOAD address in corresponding
BSAVE, if present; otherwise, loads at FROM
address in corresponding BSAVE

BRUN pathname [,LOAD=number]

- BLOADS a binary virtual diskette, then executes it
starting at first address loaded

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically
with commands in this chapter. For discussion,
see Section 3.2

[LOAD=number]

number: if decimal: between -32768 and 32767, inclusive
if hex: between \$0 and \$FFFF, inclusive

address at which memory image is to be loaded

Actual LOAD address, in order of priority:

- LOAD address of BLOAD or BRUN
- LOAD address of corresponding BSAVE
- FROM address of corresponding BSAVE

FROM=number

starting address of memory image being saved

BLOAD, BRUN, BSAVE

SIZE=number [.size-unit]

For BSAVE the amount of memory to be saved must be specified with a SIZE parameter. The size given must be less than or equal to the size that was given in the CREATE of the binary file.

Size may be given in units of bytes (characters), sectors (256 bytes each), blocks (512 bytes each), or K (1024 bytes each). The default unit for size is blocks. The size parameter has the form: SIZE=number[unit]. Unit may be 'C' for characters, 'S' for sectors, 'B' for blocks, or 'K' for K. The '.' is optional except where it is needed to resolve ambiguity in the specification of hexadecimal numbers. It is suggested that the '.' be used for readability.

For discussion, see Sections 6.1 (DOS) or 7.1 (Pascal)

[,REPLACE]

If a memory image has previously been BSAVED using the same pathname, it will not be destroyed and the present BSAVE will not be executed unless the REPLACE parameter is present. (REP is also recognized.)

Error Messages

- 51,NOT BINARY
- 52,NOT PREVIOUSLY SAVED
- 53,FROM PARAMETER REQUIRED
- 54,SIZE PARAMETER REQUIRED
- 55,FILE WAS THERE, REP NOT SPECIFIED

Access Rights Required

- R (READ) for each Cluster-file name in pathname
- W (WRITE) for file being BSAVED

Discussion

These three Cluster commands (BLOAD, BRUN, and BSAVE) allow machine language programs to run independently of DOS and Pascal. They are especially useful when customizing a BOOT program (which runs before DOS or Pascal is read in) or for running programs without making assumptions about the current operating system.

It is also possible, just as with a stand-alone Apple, to run machine language programs under DOS and Pascal using the commands already in those language systems.

Before a BSAVE is issued, a binary-type virtual diskette of sufficient size must be CREATED. Then a BSAVE is issued which stores a memory image on the virtual diskette. The user specifies the starting address (FROM=number) of the memory image and the size of the binary Cluster-file to be saved. An optional parameter, [,LOAD=number], allows the user to save a memory image from one part of memory and load it (with a BLOAD or BRUN) into a different part.

BLOAD and BRUN load a memory image previously saved by the BSAVE command. The first address loaded is the LOAD address of the BLOAD or BRUN command, if specified. Otherwise, it is the LOAD address of the BSAVE command, if specified, or the FROM address of the BSAVE, if neither LOAD address is specified.

BRUN does a BLOAD, then executes the loaded program starting at the first address loaded.

Note that it is not necessary (or possible) to MOUNT a binary virtual diskette before issuing the BSAVE, BLOAD or BRUN command.

Examples

1. Data Collection

Suppose a meteorologist uses his Apple to measure cloud patterns on satellite weather pictures. He inputs data from a picture reading device with a machine language program. To do this, he:

- a) creates a binary virtual diskette of the correct size

```
CREATE READIMAGE,T=B,SIZE=20.S
```

- b) enters and debugs his machine language code

- c) BSAVES the code on the previously created virtual diskette

```
BSAVE READIMAGE,FROM=$800,SIZE=4096.C
```

- d) executes the program with a BRUN

```
BRUN READIMAGE
```

BLOAD, BRUN, BSAVE

2. Editing in different part of memory

If the meteorologist wishes to edit his machine language program, it is sometimes convenient to read it into a part of memory different from where it will reside when executed. In the above example, he might:

- a) read the program into a different part of memory

```
BLOAD READIMAGE,LOAD=$2800
```

- b) make desired changes in the program

- c) again save the program on disk

```
BSAVE READIMAGE,FROM=$2800,SIZE=4096.C,LOAD=$800
```

- d) again run the program

```
BRUN READIMAGE
```

or, alternatively:

- c) BSAVE READIMAGE,FROM=\$2800,SIZE=4096.C

- d) BRUN READIMAGE,LOAD=\$800

CREATE pathname,type [,size] [,sectors] [,protection]

-creates a new Cluster-file with specified
type, size, passwords and access rights

Parameters and defaults:

pathname: For summary, see PATHNAME filed alphabetically
with commands in this chapter. For discussion,
see Section 3.2

type:

T=B (binary)
T=D (DOS)
T=P (Pascal)
T=Y (directory)

SIZE=number [.size-unit]

Size of the file to be created may be given in units of bytes, sectors (256 bytes each), blocks (512 bytes each), or K (1024 bytes each). The default unit for size is blocks. The size parameter has the form: SIZE=number[.unit]. Unit may be 'C' for bytes, 'S' for sectors, 'B' for blocks, or 'K' for K. The '.' is optional except where it is needed to resolve ambiguity in the specification of hexadecimal numbers. It is suggested that the '.' be used for readability.

Size is not allowed for directory (T=Y) or DOS (T=D) Cluster-files. Directories are managed by the file server and their size is not specifiable by the user. DOS files are fixed in size by the DOS system to 455 sectors for DOS 3.2.1, and 560 sectors for DOS 3.3. For binary (T=B) files, size is required. For Pascal (T=P) files, size may be given; if not, the standard Pascal disk size (280 blocks of 512 bytes each) is used. When a Pascal-type virtual diskette is created, if the number of blocks specified is not an even multiple of 8, it is rounded up to be so; the size displayed with the SHOW MOUNTS command is the true number of blocks available to the user, not the number specified by the CREATE.

For discussion, see Sections 6.1 (DOS) or 7.1 (Pascal)

[,sectors]

SECTORS=13 required for DOS 3.2.1
SECTORS=16 required for DOS 3.3

The number of sectors can only be
given for DOS-type virtual diskettes.

CREATE

[,protection] See Chapter 4.

protection: protect item [,protection]

protect item: ,GRPPW=password
,PRVPW=password
,PUBACC=access rights (Default: R)
,GRPACC=access rights (Default: PUBACC)
,PRVACC=access rights (See 4.1.3)

access rights: set of letters from RWECD (read, write, erase, create, and delete)

Error Messages

20,TYPE PARAMETER REQUIRED, (T=)
21,SIZE OUT OF RANGE (<16B or >32767B)
22,SIZE NOT ALLOWED WITH T=D OR T=Y
23,SIZE PARAMETER REQUIRED WITH T=B
24,SIZE OUT OF RANGE (<1 OR >16K)
25,T=D REQUIRES SECTORS=13 OR 16

Access Rights Required to Execute a CREATE

C (create) for parent directory
R (read) for each directory read

Discussion:

The CREATE command:

1. allocates space for specified virtual diskette or directory on specified (or default) physical disk unit
2. creates intermediate directories as necessary
3. enters its name in directory preceding it in pathname, or, if not specified, in current default directory
4. stores passwords and access rights on virtual diskette or directory being created; these are stored in a special section not accessible to the ordinary user

Directories implicitly created because they do not already exist, are given CREATE and DELETE access rights plus the access rights and passwords specified for the Cluster-file being explicitly created.

Examples

1. CREATE /1/USERS/NICK/NICK.V1,T=D,
PRVPW=GOODSTUFF,GRPACC=,PUBACC=,SECTORS=13

This example creates a virtual diskette called NICK.V1 on disk unit 1, the diskette inserted in the left-hand drive of the first system floppy. It also enters the name and address of NICK.V1 in the subdirectory NICK. (NICK is an entry in the subdirectory USERS which in turn is an entry in the root directory of the system floppy itself. All three directories and the virtual diskette are on disk unit 1.)

The created virtual diskette is of type DOS, with 13 sectors per track, so its size is automatically 455 sectors of 256 bytes, the size of a real mini-diskette in DOS 3.2 or 3.2.1.

Anyone who enters the correct pathname and the password GOODSTUFF will be allowed to read, write or erase the virtual diskette NICK.V1. Private access rights were not specified, so the default for a virtual diskette is in effect.

Users who do not know the private password are not allowed any access to NICK.V1.

2. CREATE /LANGUAGES/FRENCH/LESSON.3,T=P,SIZE=60

This example creates a virtual diskette of type Pascal whose size is sixty 512-byte blocks. The virtual diskette is called LESSON.3 and is listed in a subdirectory called FRENCH which in turn is listed in the root directory of a system floppy or hard disk called LANGUAGES.

Everyone has full access rights to this virtual diskette without giving any password: they can read it and write on it, and if they have create and delete rights for the subdirectory FRENCH, they can erase it, delete it, and then create another virtual diskette with the same name.

3. CREATE BRIDGE,T=B,SIZE=10.S

This example creates a virtual diskette called BRIDGE which is of type binary with a size of ten 256-byte sectors. Space is allocated on the physical unit specified in the current default directory, and the name and address of BRIDGE is entered in the current default directory. All users have full access rights.

4. CREATE STUDENTS,T=Y,PRVPW=SOCRATES

This example creates the directory STUDENTS on the physical unit specified in the current default directory and lists the

CREATE

name and address of STUDENTS in the current default directory. Only users giving the password SOCRATES are allowed full access rights; others are allowed to LIST the directory STUDENTS and read it when it is part of a pathname, but cannot, for example, make any new entries in the directory.

5. CREATE /COURSES/CHINESE/BEGINNING/LESSON1,T=P,PRVPW=MING,GRPPW=PEKING,PUBACC=,GRPACC=R

This example is for a new Chinese course to be stored on a system 8-inch floppy called courses which has just been formatted by the Nestar formatting utility program (see System Manager's Manual). The root directory exists, but there are no other Cluster-files on the floppy. In this example, the directories CHINESE and BEGINNING are implicitly created and the virtual diskette LESSON1 is explicitly created. There is no public access to these three Cluster-files. Users giving the private password MING are given the default private access rights: for the two directories they can create, delete, read, write and erase; and for the virtual diskette they can read, write and erase. Users giving the group password, PEKING, can only read the three Cluster-files.

DELETE pathname

-deletes a Cluster-file from system

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically with commands in this chapter. For discussion, see Section 3.2

Error Messages

29,CURRENTLY MOUNTED OR DEFAULT DIRECTORY

Access Rights Required

E (ERASE) for virtual diskette or directory being deleted
D (DELETE) for directory in which it is listed
R (READ) for all directories in pathname

Discussion

The DELETE command:

erases the specified virtual diskette or directory and removes its name and address from the parent directory

The DELETE will not be executed if the Cluster-file is currently mounted by anyone on the system or if the user does not have the required access rights, or if it is a directory which is not empty or is the current default directory of any user.

Examples

DELETE ENGLISH.1A:OLDCHAP

This example erases the virtual diskette ENGLISH.1A and removes its name and address from the current default directory.

HELP

HELP [<subject>]

-displays list of commands or syntax of specified
command or format of specified parameter

Parameters and Defaults

[<subject>]

null: lists commands

command word: gives syntax of specified command

parameter: gives format of specified parameter

CMD: gives syntax of special Pascal
drive commands (SHOW DRIVES,
SHOW SLOT, SET S<slot>,
SET D<drive>,S<slot>,
SET D<drive>,drive-status.
This is available only when
entering commands through
CMD program.

Error Message

19,HELP OPERAND NOT FOUND

Discussion

The user is allowed to enter either ? or HELP when he needs
additional information at a user-station.

Example

If the user enters:

HELP

the following message is displayed:

Enter 'HELP [<subject>]' where subject is: blood, bsave, brun,
create, delete, list, lock, unlock, mount, unmount, protect,
show, set, pathname, password, number, size, access, type,
usage, access, drive, or protection.

LIST [pathname] [,VERBOSE] [,NESTED]

-lists entries in directory

Parameters and Defaults

pathname: For summary, see PATHNAME, filed alphabetically with commands in this chapter. For discussion, see Section 3.2

[,VERBOSE]

if not present, just the names and Cluster-file types are listed

if present, the names plus other descriptive information are listed

[,NESTED]

if not present, only the directory addressed by the pathname or current default directory is listed

if present, the system lists

the addressed (or default) directory

all subdirectories whose names appear in the addressed directory

all subdirectories whose names appear in the listed subdirectories, etc. until all directories listed contain only names of virtual or system diskettes

Error Messages

109,ROOT DIR NOT SPECIFIED, NO DEFAULT

Access Rights Required

R (READ) access for each Cluster-file in pathname

Discussion

This command can be used to list the Cluster-files in a directory or a family of directories. It is possible to list either the names and types (no VERBOSE), or the names, types and sizes (VERBOSE).

If no pathname is given (e.g. LIST), then the current default

LIST

directory is listed. If the NESTED or VERBOSE parameters are given when there is no pathname, they must be separated from the word LIST by both a space and comma. For example:

```
LIST ,NESTED
LIST ,VERBOSE
LIST ,NESTED,VERBOSE
LIST
```

Examples

1. LIST /MAIN

The system will display:

```
BOOT,T=V
SYSTEM,T=Y
DOS,T=Y
PASCAL,T=Y
```

The T=Y indicates the Cluster-files which are directories; T=V indicates a virtual diskette.

2. LIST /MAIN,VERBOSE

The system will display:

```
BOOT,T=V,SIZE=11S
SYSTEM,T=Y,SIZE=3S
DOS,T=Y,SIZE=3S
PASCAL,T=Y,SIZE=3S
```

The size is given in 256 byte sectors.

3. LIST /MAIN,NESTED

```

BOOT,T=V
SYSTEM,T=Y
  ERROR,T=S
  HELP,T=Y
    HELP,T=S
    SHOW,T=S
    LIST,T=S
    BRUN,T=S
    LOCK,T=S
    MOUNT,T=S
    BLOAD,T=S
    BSAVE,T=S
    CREATE,T=S
    DELETE,T=S
    UNLOCK,T=S
    UNMOUNT,T=S
    PROTECT,T=S
    SET,T=S
    PATHNAME,T=S
    PASSWORD,T=S
    NUMBER,T=S
    SIZE,T=S
    PROTECTION,T=S
    ACCESS,T=S
    TYPE,T=S
    USAGE,T=S
    DRIVE,T=S
DOS,T=Y
  3.2.1,T=V
  PASCAL,T=Y
  BOOT,T=V

```

T=S indicates the Cluster-file is part of the Cluster system information.

4. LIST /MAIN,VERBOSE,NESTED

This is left as an exercise for the reader.

LOCK

LOCK lockname [,usage]

-locks "lockname" for exclusive (EXC) or shared (SHR) use

Parameters and defaults:

lockname:

any arbitrary name (except ALL)
(1-15 characters; no commas, slashes, colons, carriage returns or unprintable characters; imbedded blanks ok; first character any acceptable character)

usage:

EXC (exclusive)
SHR (shared)
Default: EXC

Error and Information Messages

40,ILLEGAL LOCK NAME
41,IN USE
42,IN EXC USE

Access Rights Required

None

Discussion

The LOCK command allows users at different terminals to cooperate in using system resources such as individual files or input/output devices. LOCKNAME is any arbitrary name agreed upon by more than one user. For example, if the lock is used to coordinate write access to a given DOS file, a program, before writing on that file, would issue a LOCK command. Then, if the return code is 0 indicating no other user-station is currently holding the specified lock, the program would proceed to update the file. If, however, the return code is 41 or 42 indicating the lock is currently held at another user-station, the program would go into a wait loop or perform some other task before reissuing the LOCK command.

NOTE: It is not recommended that programs enter a tight wait loop which includes a LOCK command. Instead, they should delay several seconds and then try again. This avoids tying up the system with frequent requests.

When using locks to protect a DOS or Pascal file, the user should consider the effect of buffering. When a disk write is issued, in some cases the information is transferred to the buffer but not to the disk. One way to make sure it all goes to the disk is to close the file. Buffering can also lead to problems if more than one lock is used for one file, say LOCKA for the first half and LOCKB for the second half; information at the boundary may not be handled correctly.

Therefore, the safest procedure to follow is:

1. Use only one lock for each file.
2. Close the file before issuing the UNLOCK.

Users who wish to use more than one lock per file or avoid closing the file after each write should have a detailed knowledge of exact buffering algorithms used by the current Apple DOS and Apple Pascal systems.

See Section 4.2 for a full discussion.

MOUNT

MOUNT pathname, drive [,usage]

-establishes correspondence between a virtual
diskette and drive number

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically
with commands in this chapter. For discussion,
see Section 3.2

drive:

Dd, where d=number
(1-255 for DOS; 4,5,9-12 for Pascal)

usage:

RO (read only)
RW (read/write--must have write access)
Default: RO

EXC (exclusive)
SHR (shared)
Default: SHR, if RO or not given
EXC, if RW

Error Messages

30,DRIVE REQUIRED
31,IN USE
32,IN EXC USE
33,ILLEGAL DISK TYPE
111,NO ACCESS FOR WRITE

Access Rights Required

R (read) for virtual diskette and all
directories in pathname

W (write) for virtual diskette if
being mounted RW

Discussion

The MOUNT command is analogous to inserting a real diskette in a mini-floppy disk unit attached to a stand-alone Apple. On the Cluster/One system a MOUNT command sets software pointers so that when a command is given to read or write on a given disk drive, it will address instead a virtual diskette on the central disk system.

Mounting a binary-type virtual diskette is not allowed. Instead a BLOAD or BRUN should be used.

Examples

```
MOUNT /MAIN/USERS/CAROL/LETTERS,D9,EXC,RW
```

This command mounts a virtual diskette called LETTERS, which is listed in a subdirectory CAROL, which in turn is listed in a subdirectory USERS, which in turn is listed in the root directory of a system floppy or hard disk called MAIN. The virtual diskette will be mounted on drive 9; no other user can mount LETTERS until an UNMOUNT command is issued; and Carol can both read and write the files stored on LETTERS. If someone else has already mounted LETTERS, this MOUNT will not be executed; instead an error message

31,IN USE

will be returned.

This example does not specify any passwords. Therefore, it will be executed only if:

-----no passwords or access rights were specified when the virtual diskette and directories were created; therefore, all users have all rights.

the default group or private password gives read access to all directories in the pathname and read and write access to the virtual diskette LETTERS.

OFF

OFF

-initializes user-station; BRUNs BOOT program

Discussion

In order to increase over-all system efficiency, the OFF command should be given whenever a user is through using his Apple. This frees the memory and other resources assigned to the user-station.

Giving the Cluster command is equivalent to turning power off and on at the Apple or typing H at the Pascal command level.

pathname: See Section 3.2 for a full discussion.

pathname: /unitname/filepath
or filepath

80 characters max.

An initial slash indicates first field is a unit name; no initial slash indicates current default directory is to be prefixed to pathname.

unitname: 1, 2, or 5 [:password]
or name [:password]

filepath: name[:password] [/filepath]

name: name of disk unit, directory, or virtual diskette.
(1-15 characters; may not include commas, slashes, colons, carriage returns or unprintable characters; imbedded blanks ok; first character can be any acceptable character.)

Upper and lowercase can be used for any characters within a Cluster command and are equivalent.

For certain commands a null password is allowed:

SET DIR sets current default directory
to null

LIST lists entries in current default
directory

PROTECT changes protection parameters for
current default directory

SHOW PROTECTION displays access rights for
current default directory

PROTECT

PROTECT pathname [protection]

-modifies passwords or access rights for a
virtual diskette or directory

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically
with commands in this chapter. For discussion,
see Section 3.2

[,protection] See Chapter 4 for a detailed discussion.

protection: protect item [,protection]

protect item: ,GRPPW=password
,PRVPW=password
,GRPACC=access rights (Default PUBACC)
,PRVACC=access rights (Default RWECD)
,PUBACC=access rights (Default R)

access rights: set of letters in any order from RWECD
(read, write, erase, create, and delete)

Error Messages

No errors specific to PROTECT command

Access Rights Required

Private password must be given.
No specific access rights required.

Discussion

The PROTECT command is used to change the password and access rights associated with a given directory or virtual diskette. Anyone giving the private password is allowed to use the PROTECT command, regardless of the previously assigned access rights. If there is no private password, then anyone can issue the PROTECT command.

Normally access rights and passwords are set when a directory or virtual diskette is created. The PROTECT command allows the user to change them without destroying and recreating the whole Cluster-file.

A null pathname is allowed. It refers to the current default directory. If the null pathname is used, the comma must still be present, for example:

PROTECT ,PRVPW=APRICOTS,PRVACC=RE

See Chapter 4 for a full discussion of protection.

Examples

PROTECT NANCY,PRVPW=VOLCANO,PRVACC=R,PUBACC=,GRPACC=

In this example anyone entering the password VOLCANO is allowed to read the virtual diskette NANCY, and no one else has any access rights whatsoever. There are no public or group access rights because the protect items PUBACC= and GRPACC= specify no rights. No one can erase or write on this virtual diskette until another PROTECT command is issued to assign such rights.

PROTECTION

[,protection] See Chapter 4 for a detailed discussion.

protection: protect item [,protection]

protect item: ,GRPPW=password
 ,PRVPW=password
 ,GRPACC=access rights (Default PUBACC)
 ,PRVACC=access rights (Default RWECD)
 ,PUBACC=access rights (Default R)

access rights: set of letters in any order from RWECD
 (read, write, erase, create, and delete)

SET DIR pathname

-sets current default directory

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically with commands in this chapter. For discussion, see Section 3.2

Error Messages

57,CLUSTER-FILE IS NOT A DIRECTORY

Access Rights Required

Read access for all directories in pathname

Discussion

The SET DIR command acts differently depending on whether the specified pathname begins with a slash or not. If the pathname begins with a slash, the default directory is set equal to the specified pathname. But if the pathname doesn't begin with a slash, it is appended to the existing default directory. The last name of the specified pathname must be a directory.

In any subsequent commands, the current default directory is prefixed to pathnames which do not begin with a slash. Passwords may be required in a default directory, and the user is allowed to specify them in the SET DIR command or to use the default private and group passwords set previously by the PRVPW and GRPPW commands.

Access rights for the default directory are computed when the SET DIR is issued.

A SET DIR with no pathname clears the current default directory, i.e. sets it equal to null.

The SET DIR and SET password commands can be combined; for example:

```
SET DIR /MAIN,GRPPW=PINE,PRVPW=CEDAR
```

The delimiter between DIR and pathname can be either a blank or an equal sign.

The SHOW DIR command can be used to display the current default directory (excluding passwords).

Examples

SET DIR

1. SET DIR /1/ACCOUNTING/PAYROLL

This example sets the current default directory equal to the subdirectory PAYROLL which is listed in the subdirectory ACCOUNTING which is listed in the root directory of the 8-inch floppy diskette mounted in the left drive of the system floppy disk. Now if the command is given to mount the virtual diskette SALES

MOUNT SALES,D9,RO,SHR

it is equivalent to the command:

MOUNT /1/ACCOUNTS/PAYROLL/SALES,D9,RO,SHR

2. SET DIR SUNNYVALE

Because the pathname in this example does not begin with a slash it is appended to the existing default directory. If this command is issued after the previous example, the resulting default directory would be:

/1/ACCOUNTS/PAYROLL/SUNNYVALE

SET [GRPPW=password] [PRVPW=password]

-sets default group or private password

Parameters and Defaults

password: See Section 4.1.7 for further discussion.

a secret word used to limit access to Cluster-files
(1-15 characters; may not include commas, slashes,
colons, carriage returns or unprintable characters;
imbedded blanks ok; first character can be any
acceptable character)

Error Messages

None specific to setting default passwords

Access Rights Required

None

Discussion

Having to enter a password for each Cluster-file name takes time and makes it more likely that an unauthorized user will learn the password. Therefore, the Cluster/One allows a user to enter two default passwords, one for private access and one for group access. Then if no password, or an incorrect password, is specified in a pathname, the system will see whether the private or group default password is correct, and, if so, will grant the corresponding access rights.

When a pathname is given, access will be granted under the following conditions:

The access desired is a public access right for the specified Cluster-file; either no password or an extraneous password may be specified.

The access desired is a group access right for the specified Cluster-file, and the correct group password is either specified in the pathname or stored as the default group password, or there is no group password.

SET password

The access desired is a private access right for the specified Cluster-file, and the correct private password is either specified in the pathname or stored as the default private password, or there is no private password.

The SET password and SET DIR commands can be combined; for example:

```
SET DIR /MAIN,GRPPW=PINE,PRVPW=CEDAR
```

Examples

Suppose the teacher in charge of computer assisted mathematics decides he needs a password in order to write a particular disk. All courses have the same private password: OBVIOUS. So, at the beginning of his session, he enters:

```
SET DIR /MATH
SET PRVPASS=OBVIOUS
```

Then when he mounts the virtual diskette for a given course, say ALGEBRA, instead of typing:

```
MOUNT /MATH:OBVIOUS/ALGEBRA:OBVIOUS,D9,RW,EXC
```

he can shorten it to:

```
MOUNT ALGEBRA,D9,RW,EXC
```

SHOW DATE

-displays the current date

Parameters and Defaults

None

Error Messages

None

Access Rights Required

None

Examples

SHOW DATE

The system displays the current date, for example:

07/04/80

SHOW DIR

SHOW DIR

-displays name of current default directory

Parameters and Defaults

None

Error Messages

64,NO DEFAULT DIRECTORY SET

Access Rights Required

None

Discussion

This command displays the pathname of the current default directory. Passwords supplied with a SET DIR are not displayed with SHOW DIR, but they are in effect. SHOW DIR always displays the directory pathname using a unit name rather a unit number.

Examples

SHOW DIR

The system displays the current default directory, for example:

/MAIN/USERS/ROOSEVELT

SHOW LOCK [lockname]

-lists all stations using
specified lockname

Parameters and Defaults

[lockname]

any arbitrary name (except ALL) agreed upon by cooperating users. Has the same restrictions as name: 1-15 characters; may not include commas, slashes, colons, carriage returns or unprintable characters; imbedded blanks ok; first character can be any acceptable character

Error Messages

None specific to SHOW LOCK

Access Rights Required

None

Discussion

The SHOW LOCK lockname command lists all stations currently holding the lock called lockname. The station numbers are given in hexadecimal (00 to FF).

Examples

SHOW LOCK PETER

This command will display all the station numbers which currently hold the lock PETER. For example:

04 2F 3C

At station 04, one station currently holds lock PETER. At station 2F, one station holds lock PETER. At station 3C, one station holds lock PETER. For example, if the user at station 04 enters the command SHOW LOCK PETER, the output will be:

SHOW LOCKS

SHOW LOCKS

-displays all locks set at user's own station

Parameters and Defaults

None

Error Messages

None specific to SHOW LOCKS

Access Rights Required

None

Discussion

The SHOW LOCKS command lists all locks currently held at a user's own station.

Examples

1. SHOW LOCKS

This command displays for each lock currently held at a user's own station:

the lockname

whether usage is shared (SHR) or exclusive (EXC)

the number of user stations currently holding the lock

For example, if the following is displayed:

LOCKA	SHR	2
JILL	EXC	1
DAVID	SHR	1
PETER	SHR	3

it means that one other station currently holds LOCKA for shared usage; only the user's own station holds lock JILL for exclusive usage and lock DAVID for shared usage, and two other stations hold lock PETER for shared usage.

SHOW MOUNTS

-gives information on all virtual diskettes
mounted at own station

Parameters and Defaults

None

Error Messages

None

Access Rights Required

None

Discussion

The SHOW MOUNTS command displays for each virtual diskette currently mounted:

its pathname

its drive number

whether it is read only (RO) or read/write (RW)

whether usage is shared (SHR) or exclusive (EXC)

its type

its volume number

its size given in 256 byte sectors (S) for DOS and 512-byte blocks (B) for Pascal

The size displayed is the actual size allocated. When a Pascal-type virtual diskette is created, if the number of blocks specified is not an even multiple of 8, it is rounded up to be so. Therefore, the size displayed with the SHOW MOUNTS will show the true number of blocks, not the number specified by the CREATE. SHOW MOUNTS always displays the directory pathname using a unit name rather a unit number.

The Cluster/One system uses some sectors for housekeeping information, but these sectors are not counted when the SHOW MOUNT command displays size. However, the LIST command displays a size which is one sector more than that shown by SHOW MOUNTS.

SHOW MOUNTS

Examples

SHOW MOUNTS

The system will display:

NAME	DRV	ACCESS	T	VOL	SIZE
/MAIN/PASCAL/LIB					
	5	RO SHR	P	1	1200B
/MAIN/USERS/EPS2					
	4	RW EXC	P	1	280B

Volume numbers are assigned by the user to each DOS virtual diskette during the INIT command; the default is 254. Pascal virtual diskettes are always assigned number 1 by the Cluster/One system.

SHOW PROTECTION [pathname]

-displays group, private and public access
rights for Cluster-file addressed by pathname

Parameters and defaults:

pathname: For summary, see PATHNAME, filed alphabetically
with commands in this chapter. For discussion,
see Section 3.2

Error Messages

64,NO DEFAULT DIRECTORY SET

Access Rights Required

R (READ) access for every Cluster-file in pathname

Discussion

This command can be abbreviated to

SHOW PROTECT [pathname]

It displays public, group and private access rights for the
specified Cluster-file (or default directory if no pathname is
specified). Passwords are never displayed.

Example

SHOW PROTECT /MAIN/USERS/NSS/2

When this virtual diskette was created, no protection parameters
were specified. The system responds:

PUBACC=R
GRPACC=R
PRVACC=RWE

SHOW STATION

SHOW STATION

-displays the number of the user station

Parameters and Defaults

None

Error Messages

None

Access Rights Required

None

Examples

SHOW STATION

The system displays the hexadecimal user station number, for example:

07

SHOW TYPE drive

-displays type of virtual diskette mounted
on drive d

Parameters and Defaults

drive

Dd, where d=number
(1-255 for DOS; 4,5,9-12 for Pascal)

Error Messages

61,DRIVE REQUIRED
62,NOT CURRENTLY MOUNTED
65,TYPE=PASCAL
66,TYPE=DOS
69,UNKNOWN TYPE

Access Rights Required

None

Discussion

This command is used to determine the type of a given Cluster-file. Its main use is by BOOT programs which must decide whether to call Pascal or DOS when a given volume is booted.

Examples

SHOW TYPE D4

The system responds:

65,TYPE=PASCAL

SHOW VOLS

SHOW VOLS

-displays the currently active physical disk drive names and numbers.

Parameters and Defaults

None

Error Messages

None

Access Rights Required

None

Examples

SHOW VOLS

The system displays the current disk drives, for example:

DISK UNIT 1 IS /SPARE
DISK UNIT 5 IS /MAIN
0,OK

UNLOCK lockname or UNLOCK ALL

-releases locked usage on a lockname
or on all locknames currently held

Parameters and Defaults

lockname

any arbitrary name agreed upon (except ALL)
(1-15 characters; no commas, slashes, colons,
carriage returns or unprintable characters;
imbedded blanks ok, first character any
acceptable character)

Error Messages

45,NOT HELD

Access Rights Required

None

Discussion

The UNLOCK command frees a lock on an agreed upon lockname, so that users at other terminals know it is all right now to access a file, external device, or other Cluster/One resource.

Examples

1. UNLOCK AMY

This command unlocks the lock called Amy.

2. UNLOCK ALL

This command unlocks all locks currently held at a user's own station.

UNMOUNT

UNMOUNT drive or UNMOUNT ALL

-cancels previous mount(s)

Parameters and Defaults

drive

Dd, where d=number (either \$hex or decimal)
(1-255 for DOS; 4,5,9-12 for Pascal)

Error Messages

35,DRIVE or "ALL" REQUIRED
36,NOT CURRENTLY MOUNTED

Access Rights Required

None

Discussion

It is not necessary to do an UNMOUNT before mounting the same or a different virtual diskette on a given drive. UNMOUNT is useful primarily to protect a virtual diskette mounted with read/write access from accidental writes after it is no longer needed.

Warning: if a Pascal user issues a command to

UNMOUNT ALL

the Pascal system on that Apple will crash as there is no booted volume on line, and it will be necessary to turn power off and on to recover. Therefore, UNMOUNT ALL should never be given in Pascal, unless immediately followed by appropriate reMOUNTs.

Examples

UNMOUNT D9

This command resets the pointers which caused disk commands to read and write to a specified virtual diskette whenever drive 9 was addressed.

Chapter 6

Using DOS with the Cluster/One System

DOS 3.2, 3.2.1, and 3.3 can run on the Cluster/One system. If there is no Language Card in a given Apple, either Integer BASIC or Applesoft BASIC can run, whichever is in the Apple ROM. If there is a Language Card, the BOOT program loads the type of BASIC not in ROM into the Language Card, so that both are available at all times. (Enter FP for floating point Applesoft or INT for Integer BASIC.)

6.1 Size and Number of Sectors

Size cannot be specified when a DOS-type virtual diskette is created; it is always the same size as a real 5 1/4 inch floppy. However, for DOS-type virtual diskettes the user must specify the number of sectors per track (13 or 16), and this affects the total number of sectors in the virtual diskette created.

<u>Field</u>	<u>Operating System</u>	<u>Size Created</u>
SECTORS=13	DOS 3.2 or 3.2.1	456 sectors
SECTORS=16	DOS 3.3	561 sectors

One of the sectors is used for Cluster information, so the number of sectors available to the user are 455 and 560 sectors, respectively.

6.2 Drives

On a stand-alone Apple, DOS allows only drives 1 and 2 to be specified, but the Cluster/One system allows any drive number from 1 to 255. To enable this capability, it is necessary to POKE the number \$FF (255) into the DOS word containing the maximum drive number. This should be done either in the HELLO program, which is executed when a virtual diskette is booted, or near the beginning of each application program. The address is different for different size memories:

Memory Size	Hexadecimal Address	Decimal Address
48K	\$A95B	43355 (-22181)
32K	\$695B	26971
16K	\$295B	10587

The above addresses are the same for DOS 3.2, 3.2.1 and 3.3.

Before poking, a preliminary PEEK to see that the present value is 2 will verify that the correct address has been selected.

Warning: This change should not be made unless more than two drives are required. If an Apple user-station has no standard Apple disk control card, drive numbers greater than 2 will cause no problem. But if there is a standard Apple disk control card and a user erroneously gives a drive number greater than 2 in a command to a real mini-disk, DOS will no longer catch the error and results are unpredictable.

6.3 Using Immediate Execution Mode in DOS

See Section 2.5 for a hands-on tutorial. The following is a brief summary:

1. DOS presents the prompt, > for INT or] for FP.
2. Enter PR#n where n is the number of the slot containing the network interface card, typically 6:

PR#6

3. DOS again presents > or]
4. Enter an @ sign followed by a Cluster command:
e.g.

@LIST

5. A return code number and message is displayed on the screen; this may be either an error message or an information message, e.g.

0,OK

1,ILLEGAL COMMAND

66,TYPE=DOS

6. Steps 3, 4, and 5 may be repeated indefinitely.

6.4 Using Cluster Commands within a BASIC Program

Different BASIC commands are used to communicate with the disk server, depending on whether Integer or Applesoft BASIC is being used. The differences are due to the fact that CHR\$ is not available in Integer BASIC and the INPUT command behaves somewhat differently in the two languages.

6.4.1 Applesoft BASIC

The following Applesoft program segment sends a Cluster command to the disk-server and reads the code and message it sends back. It is assumed that DOS is present and operational.

```
10 CD$="any Cluster Command"
```

Sets the string variable CD\$ equal to a string consisting of a Cluster command

```
20 PRINT CHR$(4);"PR#6"
```

CHR\$(4) is a ctrl-D. It signifies that the next field is a DOS command. "PR#6" enables the network card to look at all subsequent print statements to see whether they contain a Cluster command to ship to the disk-server.

```
30 PRINT CHR$(14),CD$
```

CHR\$(14) is a ctrl-N. It signifies that the next field is a Cluster command.

```
40 PRINT CHR$(4); "IN#6"
```

CHR\$(4) again signifies a DOS command. "IN#6" tells the following INPUT command to take its input from slot 6 rather than from the keyboard.

```
50 INPUT NN,NN$
```

The disk-server sends its return code (NN) and message (NN\$) to the network card in slot 6. After reading the last character from slot 6, the INPUT command resets the input slot so the next input will come from the keyboard.

Note: The "PR#6" and "IN#6" assume the network card is in slot 6. If the card were actually in slot 7, the commands would be "PR#7" and "IN#7". The exact slot must be specified.

See Section 2.7 for a greeting program which uses Cluster commands with Applesoft BASIC.

6.4.2 Integer BASIC

The same program segment written in Integer BASIC follows. This program, unlike the Applesoft program above, prints an error message if the Cluster command cannot be executed. It is assumed that DOS is present and operational.

```
10 DIM RET$(50), CD$(120)
20 N$="": REM THERE IS AN INVISIBLE
   CTRL-N BETWEEN THE QUOTES
30 D$="": REM THERE IS AN INVISIBLE
   CTRL-D BETWEEN THE QUOTES
40 CD$="Any Cluster command"
50 PRINT D$;"PR#6"
```

D\$ is a ctrl-D; it signifies that the next field is a DOS command. PR#6 enables the network card to look at all subsequent print statements to see whether they contain Cluster commands.

```
60 PRINT N$;CD$
```

N\$ is a ctrl-N; it tells the network card that the following field is a Cluster command to be sent to the disk server.

```
70 PRINT D$;"IN#6"
```

D\$ again signifies a DOS command. "IN#6" tells the following INPUT command to take its input from slot 6 rather than from the keyboard.

```
80 INPUT RET$
```

The disk-server sends its return code number and message (separated by a comma) to the network card in slot 6. After reading the last character from slot 6, the INPUT command resets the input slot so next input will come from the keyboard.

```
90 IF RET$#"0,OK" THEN PRINT  
  "ERROR ";RET$
```

This command prints any
return code and message except
"0,OK"

6.5 Initializing a Virtual Diskette in DOS

See Sections 2.6 and 2.7. The examples given are for Applesoft BASIC. If Integer BASIC is to be used, use the above example, to modify the CHR\$ and INPUT commands and add a dimension statement.

NOTE: You must always INIT a 16-sector virtual diskette under DOS 3.3 and INIT a 13-sector virtual diskette under DOS 3.2.1. However, it is possible to both read and write these two virtual disk formats from either DOS under Cluster/One's File Server. You need not, therefore, convert all DOS 3.2.1 virtual diskettes to DOS 3.3 in order to use DOS 3.3. This is not true for real Apple mini-diskettes.

Chapter 7

Using Pascal with the Cluster/One System

The Cluster/One Model A runs with Apple Pascal Version 1 (U.C.S.D. Version II.1). In order to use Pascal on a given user-station, that specific Apple must contain a Language Card in slot 0.

Essentially, the Pascal Operating System runs in the Cluster/One system just as it does with a stand-alone Apple. However, there are a few minor differences which will be covered in this chapter:

- virtual diskettes of variable size
- virtual and real disk drives
- issuing Cluster commands with the CMD program
- issuing Cluster commands within a user's program.

7.1 Virtual Diskettes of Variable Size

Size is an optional parameter when CREATEing a Pascal-type virtual diskette. If no size is specified, each Pascal-type virtual diskette is, by default, the size of a real mini-diskette: 280 blocks, each 512 bytes long.

Size may be specified with the CREATE command using the parameter:

[,SIZE=b]

where b is the number of 512 byte blocks. The maximum size is limited by the size of the disk unit on which the virtual diskette will be stored. The number of blocks for a Pascal-type virtual diskette must be:

>=16	(minimum)
<=1200	(max for 8-inch system floppy)
<=32767	(max for system hard disk)

Because there are eight blocks per simulated mini-disk track, the number of blocks is always rounded up to the nearest multiple of eight. The extra blocks are available for use. The true number of blocks assigned is displayed by the SHOW MOUNTS command.

NOTE: The system allows size to be specified in units of:

C	(characters--1 byte each)
S	(sectors--256 bytes each)
B	(blocks--512 bytes each)

PASCAL

blank (blocks--512 bytes each)
K (kilos--1024 bytes each)

Numbers may be given in \$hex or decimal. For details, see the CREATE command writeup in Chapter 5.

7.2 Drives

Each user-station operating under Pascal has 6 mini-disk drives (4, 5, 9-12) which can be either real or virtual. Real drives refer to a private mini-disk attached directly to the user-station; virtual drives refer to software changes which cause a disk read or write command to address a virtual diskette on the central disk system rather than a private mini-disk drive.

The six drives available in Pascal are divided into three drive pairs: 4/5, 9/10, and 11/12. Both drives of a drive pair must have the same drive-status, i.e., be either real or virtual to the same slot.

The BOOT program, part of the Cluster/One System software, sets the drive status of each drive-pair when power is turned on at an Apple user-station. Whether a drive-pair is real or virtual depends on the presence or absence of network interface cards or standard Apple DOS 3.2 or Pascal Disk II controllers in certain slots. At power-on the drive-status for each pair is as follows:

<u>Drive-Pair</u>	<u>Drive-Status</u>
4/5	Always virtual to booted slot (typically 6 but can be 1-7. Pascal restrictions on the use of slots must be considered.)
9/10	Real if slot 4 contains a standard Apple disk controller Virtual to slot 4 if slot 4 contains a network interface card Virtual to booted slot if slot 4 is empty
11/12	Real if slot 5 contains standard Apple disk controller Virtual to slot 5 if slot 5 contains a network interface card Virtual to booted slot if slot 5 is empty

In the normal case where the network interface card is in slot 6 and slots 4 and 5 are empty, all three drive-pairs are virtual to slot 6.

BOOT boots from the highest numbered slot containing a network card (normally 6) and looks for second and third network cards in slots 4 and 5.

If multiple network interface cards are present, and the secondary (or tertiary) nets are not running, Pascal will try to read disk labels from these drives, and may hang waiting for a response. In this situation typing ctrl-shift-N will permit things to proceed. In fact, if only one network interface card is present, and that net is temporarily not running, entering ctrl-shift-N will permit local processing to continue at the user-station.

The CMD program described in the following sections can be used to override the above defaults to anything else suitable. Commands to set and display slots and drive-status are given in Sections 7.3.1 to 7.3.5.

7.3 CMD

The CMD program is an interactive program supplied by Nestar which allows a user to enter Cluster commands at his keyboard. CMD is generally made available to you as a CODE file in the default library automounted on drive 5 by BOOT. If not found there, the System Manager will know where it is installed.

The CMD program sends each command as entered to the disk-server (except for the pseudo commands described below). The disk-server executes the command, if possible, and, in any case, returns an integer value and message indicating the status of the command. The message is then displayed by CMD.

CMD again displays its prompt (:) for another Cluster command to be entered. The cycle repeats until a carriage return is entered; then control returns to the Pascal command level.

The following print-out shows a sample run. User responses are shown in lower case to distinguish them from information displayed by Pascal and CMD. (Actually, commands can be entered in either upper or lower case, and output from Pascal on an Apple with upper/lower case capability is usually upper and lower case.)

PASCAL

```
COMMAND: E(DIT,R(UN,F(ILE,C(OMP,L(INK,X(ECUTE,A(SSEM,D(EBUG,?
x
EXECUTE WHAT FILE?#5:cmd
ENTER NETWORK COMMAND

:create lunch
ROOT DIR NOT SPECIFIED AND NO DEFAULT DIR

:SET DIR /MAIN/PASCAL
OK

:SHOW DIR
/MAIN/PASCAL
OK

: (return)
COMMAND: E(DIT,R(UN,F(ILE,C(OMP,L(INK,X(ECUTE,A(SSEM,D(EBUG,?
```

In addition to the Cluster commands which it sends to the disk-server, CMD responds to a number of pseudo-commands which it executes itself. These commands are concerned with displaying and setting the drive-status (real or virtual) and network-interface-card slot associated with each Pascal virtual drive.

7.3.1 SHOW DRIVES

This command displays the status (real or virtual) for each drive pair; it also displays the slot containing the network interface card associated with each virtual drive pair. For example, when a Pascal disk has just been booted, and slot 6 contains the network interface card and slots 4 and 5 are empty, the SHOW DRIVES command displays the following:

DRIVE-PAIR	STATUS(SLOT)
4/5	VIRTUAL(6)
9/10	VIRTUAL(6)
11/12	VIRTUAL(6)

OK

7.3.2 SET D<drive>,drive-status

This command sets the specified drive (and the other member of the drive-pair) to either real or virtual through the default slot number. The parameter <drive> must be 4, 9, or 11; and drive status must be one of the following keywords: R, V, REAL, or VIRTUAL.

SET D is useful when a physical floppy is present and the user wishes to temporarily override its drive to virtual.

7.3.3 SHOW SLOT

This command displays the default slot number, which should be the slot containing the network interface card. This value is used in two ways: to set the slot for virtual diskettes and when sending commands to the disk server. It is set by the SET S<slot> command. If a slot number is set to a slot which does not contain a network card, CMD will print out "No Network Card in Slot" the next time it sends a command to the disk server.

The default slot number is used only by the CMD program.

A SHOW SLOT command issued after the SHOW DRIVES command above will display:

DEFAULT SLOT = 6

7.3.4 SET S<slot>

This command sets the default slot number. The parameter <slot> must be 1-7. Unless more than one network card is installed, this command should not be used.

7.3.5 SET D<drive>,S<slot>

This command sets the specified drive to virtual through the interface card in the specified slot number. The parameter <drive> must be 4, 9 or 11; <slot> must be 1-7. If a drive is set virtual to a slot that does not contain a network interface card, unpredictable results will occur. Unless more than one network card is installed, this command should not be used.

7.4 Using Cluster Commands Within a Pascal Program

Nestar provides for each Pascal-capable user-station the following software:

CMD.CODE

NETUNIT.LIBRARY, a library containing:

NETUNIT.CODE, a Pascal regular unit containing the function SENDMSG

NETTALK.CODE, an assembly language procedure used by SENDMSG.

PASCAL

The CMD program is the interactive program described in Section 7.3. It uses the Pascal function SENDMSG which in turn uses the assembly language procedure NETTALK. NETTALK calls the machine language routines stored in ROM on the network interface card.

If a user desires to execute Cluster commands within his own program, he can use SENDMSG, just as the CMD program does. SENDMSG is a Pascal function which sends Cluster commands to the disk-server and accepts the code and message which the disk-server returns. The function has the following form:

```
RC:=SENDMSG(NETID,COMMAND,RETMSG)
```

where:

RC is an integer representing the "return code" sent back by the file-server--e.g. 0 for OK; 1 for ILLEGAL COMMAND

NETID is an integer whose value indicates which slot contains the Nestar interface card. The values 4-7 indicate an actual slot number; 0 indicates the function should search for a slot containing a network interface card. The search checks each of the drive-pairs (4/5, 9/10, 11/12) in ascending order. The slot number of the first pair that is virtual is used by the SENDMSG function.

COMMAND is a string variable whose value is the Cluster command to be executed

RETMSG is a string which will contain the return message, such as OK or IN USE, which corresponds to RC, the return code number.

SENDMSG is part of NETUNIT.LIBRARY, a Pascal library supplied by Nestar. It may be installed in SYSTEM.LIBRARY by a local installation. NETUNIT calls the assembly language procedure NETTALK which is also part of NETUNIT.LIBRARY. SYSTEM.LIBRARY is usually stored on the Pascal virtual diskette mounted on drive 5.

If not installed in SYSTEM.LIBRARY, the Pascal program should use the \$U option to specify the location of NETUNIT.LIBRARY to the compiler. During linking, the same name must be given.

To compile a program using SENDMSG, the following steps can be taken:

1. NETUNIT and NETTALK can be included in SYSTEM.LIBRARY.
2. SYSTEM.LIBRARY can be on the booted diskette.
3. The compiled program can be linked using SYSTEM.LIBRARY before it is run. (This occurs automatically if the compiled program is stored on the workfile.)

In Apple Pascal, the NETUNIT interface is defined as follows:

```
UNIT NETUNIT;

INTERFACE

TYPE NETMSG = STRING [255];

FUNCTION SENDMSG (NETID: INTEGER; COMMAND: NETMSG;
  VAR RETMSG: STRING): INTEGER;
```

The following program, MINICMD, demonstrates the use of NETTALK, NETUNIT and SENDMSG within a Pascal program. (SENDMSG calls NETTALK; it is not explicitly mentioned.) MINICMD is a bare-bones version of the CMD program which is used for immediate execution of Cluster commands.

MINICMD displays:

ENTER CMD:

and sends to the disk server whatever is typed. The disk-server executes the command, if possible, and returns a number code and message, which MINICMD displays. This sequence is repeated until the user responds with a carriage return alone.

MINICMD differs from CMD in the following ways:

1. Both the code number and reply returned by file-server are displayed on the console.
2. The commands referring to the user's own drives (SHOW DRIVES, SHOW SLOT, SET Sn, and SET Dd, drive-status) are not available in MINICMD. They are executed by the CMD program, not the disk-server.

```

PROGRAM MINICMD;

USES NETUNIT;

VAR CMD,REPLY:STRING;
    RC: INTEGER;

BEGIN
WRITELN(' MINI COMMAND PROGRAM');
REPEAT
    WRITE('ENTER CMD:');
    READLN(CMD);
    IF LENGTH(CMD)<> 0 THEN
        BEGIN
            RC:=SENDMSG(0,CMD,REPLY);
            WRITELN(RC,',',REPLY);
        END;
    UNTIL LENGTH(CMD)=0;
END.

```

7.5 Initializing a Virtual Diskette in Pascal

Suppose that a brand new user wants to write new Pascal programs or run existing ones on the Cluster/One Model A system. What must he do to get started?

In general, he (or the system manager) must:

boot an existing Pascal virtual diskette.

CREATE a Pascal-type virtual diskette using CMD.

MOUNT it on drive 9, 10, 11, or 12.

Zero its directory (See Apple Pascal Reference Manual, p.38).

Transfer SYSTEM.MISCINFO and SYSTEM.PASCAL from the booted virtual diskette to the virtual diskette on drive 9, 10, 11, or 12.

Boot the new virtual diskette.

Decide whether to write a customized SYSTEM.STARTUP program which will be executed whenever the virtual diskette is booted, or to use the standard Pascal startup routine which displays a standard welcome message and the current date.

Use the Filer to transfer files from a real diskette inserted in a mini-disk attached to the user station,

if desired.

Use the Editor, Compiler, and Linker to create Pascal programs, just as on a stand-alone Apple.

For example, say that an educator has a set of arithmetic courses which he previously ran on a stand-alone Apple. Now he wants to present these same courses on the Cluster/One Model A.

Let's assume there is a system 8-inch floppy called COURSES and that he wants to CREATE a virtual diskette called ARITHMETIC on it. He would give the CREATE command:

```
CREATE /COURSES/ARITHMETIC,T=P
```

(Assume that no protection is desired and that the size of a real mini-diskette is satisfactory.)

Then he would mount the virtual diskette on drive 9:

```
MOUNT /COURSES/ARITHMETIC,RW,D9
```

Then using the Pascal FILER, he would zero the directory on the virtual diskette, transfer SYSTEM.PASCAL and SYSTEM.MISCINFO from #4: to #9: and transfer files from the real diskette on to his virtual one.

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

THE LITERATURE OF THE AMERICAN WEST

Appendix A

Error and Information Messages

When the file-server receives a Cluster command from a user-station, it executes the command, if possible, and, in any case, returns two variables:

a return code--an integer between 0 and 140

a return message, such as OK, ILLEGAL COMMAND, or TYPE=PASCAL

Chapters 6 (DOS) and 7 (PASCAL) describe how these variables are displayed in direct mode under DOS or by the Pascal CMD program and how to access them within a BASIC or Pascal program.

Messages are of several types:

information, such as OK, IN USE, TYPE=BINARY

user errors such as syntax errors or attempting to access a Cluster file without the necessary access rights

system errors such as memory or disk full; these should be reported to the system manager.

In this list, messages are classified as:

general syntax

pertaining to a specific command

system errors

file system errors.

ERRORS

General Syntax (0-19)

- 1, ILLEGAL COMMAND
- 2, NAME PARAMETER REQUIRED
- 3, UNRECOGNIZED PARAMETER
- 4, ILLEGAL DRIVE NUMBER
- 5, ILLEGAL STATION NUMBER
- 6, STATION NUMBER MUST BE HEX (USE \$NN)
- 8, ILLEGAL TYPE PARAMETER
- 9, COMMAND TOO LONG
- 19, HELP FILE NOT FOUND

Create (20-25)

- 20, TYPE PARAMETER REQUIRED (T=)
- 21, SIZE OUT OF RANGE (<16B OR >32767B)
- 22, SIZE NOT ALLOWED WITH T=D OR T=Y
- 23, SIZE PARAMETER REQUIRED WITH T=B
- 24, SIZE OUT OF RANGE (<1C OR >48K)
- 25, T=D REQUIRES SECTORS=13 OR 16

Delete (26-29)

- 29, CURRENTLY MOUNTED OR DEFAULT DIR

Mount (30-34)

- 30, DRIVE REQUIRED
- 31, IN USE
- 32, IN EXC USE
- 33, ILLEGAL DISK TYPE

Unmount (35-39)

- 35, DRIVE OR "ALL" REQUIRED
- 36, NOT CURRENTLY MOUNTED

Lock (40-44)

- 40, ILLEGAL LOCK NAME
- 41, IN USE
- 42, IN EXC USE

Unlock (45-49)

- 45, NOT HELD

Bsave/Bload/Brun

- 51, NOT BINARY
- 52, NOT PREVIOUSLY SAVED

53, FROM PARAMETER REQUIRED
 54, SIZE PARAMETER REQUIRED
 55, FILE WAS THERE, REP NOT SPECIFIED

Set/List (56-59)

57, CLUSTER-FILE IS NOT A DIRECTORY

Show (60-69)

60, SHOW OPTION NOT RECOGNIZED
 61, DRIVE REQUIRED
 62, NOT CURRENTLY MOUNTED
 64, NO DEFAULT DIRECTORY SET
 65, TYPE=PASCAL
 66, TYPE=DOS
 67, TYPE=BINARY
 68, TYPE=DIRECTORY
 69, UNKNOWN TYPE

System Errors (90-99)

95, UNRECOGNIZABLE NETWORK RESPONSE
 96, NO NETWORK CARD IN SLOT
 97, MEMORY FULL
 98, NETWORK ERROR
 99, USER ABORT

Cluster-File Errors (101-133)

101, END OF FILE OR RECORD OUT OF RANGE
 102, CLUSTER-FILE NOT FOUND
 103, BAD DELIMITER IN PATHNAME
 104, FILE NAME OR PASSWORD TOO BIG
 105, NO ACCESS FOR READ TO DIRECTORY
 106, NON DIRECTORY FOUND IN PATHNAME
 107, END OF PATHNAME IS A DIRECTORY
 109, ROOT DIR NOT SPECIFIED, NO DEFAULT
 110, NO ACCESS FOR READ
 111, NO ACCESS FOR WRITE
 112, NO ACCESS FOR APPEND (not yet implemented)
 113, NO ACCESS FOR ERASE
 114, NO ACCESS FOR CREATE
 115, NO ACCESS FOR DELETE
 116, CANT DELETE NON-EMPTY DIRECTORY
 117, MEMORY FULL
 118, DISK FULL
 119, DIRECTORY FULL
 120, INTERNAL ERROR
 121, UNINITIALIZED DISK
 122, WRONG SOFTWARE VERSION
 123, CLUSTER-FILE ALREADY EXISTS

ERRORS

124,DISK I/O ERROR
125,VERIFY FAILED (BAD MEMORY)
126,DISKETTE WAS CHANGED
127,BAD DISKETTE FORMAT
128,BAD UNIT NUMBER IN PATHNAME
129,DISKETTE WRITE PROTECTED
130,NEED PRIVATE PASSWORD FOR PROTECT
131,UNIT NAME NOT FOUND
133,DISK NOT READY

Appendix B

The BOOT Program

B.1 Description

The BOOT program is a standard part of the Cluster/One Model A File Server, and provides a means for users to startup Apple DOS 3.2, Apple DOS 3.3, or Apple Pascal Version 1. It is automatically loaded and run when an Apple with the autostart ROM, and a Nestar network interface card, is powered up.

The BOOT program is a machine language program, which executes in the memory area \$800 to \$11FF. It has been provided as a BSAVED virtual Cluster-file, in the file /MAIN/BOOT. This section describe the action of BOOT as distributed by Nestar. You should consult your System Manager to see if he has made any local modifications to the version on your Nestar File Server.

BOOT is brought into a user station's memory and executed whenever the Apple Autostart ROM (present on Apple II Plus, Apple II with Language Card or Integer machines which have upgraded by installing the ROM) recognizes a power on condition. On machines which do not have the Autostart ROM, you must explicitly enter

Cn00G (n is the slot in which
a Nestar network interface
card is installed)

to the Apple II Monitor prompt ("*"). Entering PR#6 or 6ctrlP will NOT cause the BOOT program to be run. This sequence differs from Apple Disk II controllers, which will boot whenever any character is printed to the slot they are plugged into.

When a user station is powered up BOOT will automatically unmount all disks in use by that station, free any locks held, and reset any default directory or passwords to nil. This also happens when the OFF Cluster command is issued (or when the machine is turned off and back on again). It is strongly recommended that this be done by every user who is finished using an Apple on the Model A system. This gives a clear indication to others that the station is not in use, and also frees up resources other stations may need.

B.2 Operation of BOOT

BOOT will display on the Apple screen the Nestar logo,

BOOT

followed by:

CLUSTER/ONE MODEL A NETWORK BOOT 1.0

STATION ADDRESS XX

VOLUME TO BOOT: /MAIN/

where XX is your station address (in hex, from 00 to 7F). One should place a sticker with the station number on each user station, and you should observe that the number shown on the screen agrees. If not, please consult your System Manager.

NOTE: Users who have a Sup'R'Term 80 column card for use in Pascal will find that the BOOT program recognizes that card, and displays its output using the screen attached to the serial card. The use of the Sup'R'Term card under DOS and the Model A is not especially convenient, since commands are routed to the Nestar File Server using PR#n (n is the slot in which the Nestar card is installed), while the Sup'R'Term card only displays output when a PR#3 is performed.

The user may now enter the volume name of a DOS or PASCAL disk he wishes to load his system from. This name may be up to 40 characters long, including the prefix. If the volume is not on the default prefix which is presented (shown above as /MAIN/), the user may backspace and edit this input exactly as if he had typed it himself.

You should note that the use of the Model A Nestar File Server does not in any way modify the minimum features necessary to load and execute DOS or Pascal. Any errors you would get from these standard systems will still be reflected in the Model A system. For example, you cannot boot a Pascal system into a machine which does not have a Language Card. You cannot run a DOS slave diskette, which was initialized on a 48K Apple on a 32K Apple. Certain things are done for you automatically, as a convenience to the user, wherever feasible. These are discussed in the following paragraphs.

After you have entered the name of the volume you wish to boot, the BOOT program attempts to locate that volume, and issue a MOUNT command for drive 1. If successful, the type of the disk is checked, and DOS is distinguished from Pascal.

B.2.1 Booting DOS Virtual Diskettes

For a DOS disk, the system reads the first sector, and determines whether this disk appears to be a DOS 3.2 (i.e.

13 sector disk) or a DOS 3.3 (i.e. 16 sector disk), at which time it begins the bootstrap process in a fashion parallel to the action of booting a physical minidisk. The DOS is loaded, relocated if a master disk, and then started. Any HELLO program will be autorun, etc. as is standard.

As a convenience to the DOS user, BOOT checks to see if a Language Card is present, and makes an attempt to load the Language Card with whichever version of BASIC is not present in ROM. Unless changed by the installation, this is done by locating a Pascal volume, called /MAIN/PASCAL/LIB, and locating the files FPBAS.DATA or INTBAS.DATA. (These files should have been installed by your System Manager during installation.)

B.2.2 Booting Pascal Virtual Diskettes

In the case of a Pascal disk, a check is made to see that the Language Card is present before attempting to load Pascal. If present, the Pascal volume /MAIN/PASCAL/LIB is located, and mounted RO and SHR on drive 5 of the users station. The volume the user requested to boot is remounted on the users drive 4, for RW and EXC usage.

No message will be presented should the mount for PASCAL/LIB fail. However, if the mount is successful, the file SYSTEM.APPLE must be present in that volume, or an error message will be presented. If PASCAL/LIB cannot be mounted, SYSTEM.APPLE must be present on the booted volume, now on drive 4.

Once this is done, and SYSTEM.APPLE is loaded into the Language Card, the Pascal system is started. The booted disk MUST contain the file SYSTEM.PASCAL and SYSTEM.MISCINFO, with the appropriate GOTOXY installed for the type of terminal the user is using.

Pascal will now come up, and follow its normal procedures. It will recognize a serial or communications card in slot 3, and direct its input/output to that card from now on.

Appendix C

NESTAR DISK COPY Utility ProgramC.1 Description

NESTAR DISK COPY provides an Apple user the ability to copy the contents of an entire diskette using one Apple drive, two Apple drives or between an Apple drive and the Nestar Cluster/One Model A. NESTAR DISK COPY will copy BASIC or Pascal diskettes using 13 or 16 sectors per track formatting. NESTAR DISK COPY relies on Read-Write-Track-Sector (RWTS), the Disk Operating System's routine responsible for reading and writing individual sectors and for formatting all 35 tracks of a new diskette.

NESTAR DISK COPY is distributed on DOS volumes as part of the standard Cluster/One Model A File Server system. Ask your System Manager the identification of the volumes on which this program resides at your installation.

C.2 General Operation

A) Boot a BASICs diskette that has the same sector per track formatting (13 or 16) as the intended original diskette to be copied. If you intend to copy a 13 sector formatted diskette, you must boot DOS 3.2 or 3.2.1 system. For copying a 16 sector diskette, either a 16 sector BASIC or Pascal disk, be sure to boot a DOS 3.3 diskette.

B) If you intend to use a single Apple drive it is suggested that you next type:

MAXFILES 1

C) If you are going to use any Nestar Cluster/One virtual disks in the copy procedure (including the disk on which DISK COPY resides) then ensure that they are currently mounted on the desired drives.

Note that the Nestar File Server supports virtual drive numbers 1 through 255, so you may have more than just drives 1 and 2 available for multiple disk copies, etc.

Also, be sure that Nestar volumes are mounted RW, if you want to copy to them. Further, they must be mounted EXC if

DISK COPY

you intend to INIT them during the copy process.

You can now enter

BRUN NESTAR DISK COPY [,Sn,Dd]

The program displays the following information:

```
NESTAR DISK COPY 1.0    DOS/16 <-- 13 or 16 sectors
                        <-- DOS (vtoc) or Pascal (no vtoc)
ORIGINAL                COPY
(REAL)                  (VIRTUAL) <-- REAL=Apple, VIRTUAL=Nestar

SLOT=4                  SLOT=6    <-- numbers entered by user
DRIVE=1                 DRIVE=99
VOLUME=1                VOLUME=123

READING                 <-- reading or writing

TRACK=11 SECTOR=0F      <-- track and sector currently
                        being copied

PRESS RETURN TO CONTINUE <-- waiting for user to insert
                        original or copy diskette
```

D) NESTAR DISK COPY will request both numeric (decimal) and textual input from the user. For text input, just press a single character such as Y or N if the program is requesting a Yes or No response. The <RETURN> key is unnecessary. For numeric (decimal) input, NESTAR DISK COPY allows you to edit any screen characters using CTRL-X (delete current line), -> (advance), or <- (backspace). Once you have completed entering and editing your response, press the <RETURN> key.

At any time, except during INITing, you can press the <ESC> key to exit the program. If you press <ESC>, NESTAR DISK COPY will abort the copying and allow you to restart.

C.3 Specific Operation

A) Copying standard (with VTOC) BASIC diskettes. (DOS 3.2, 3.2.1, 3.3 -- 13 or 16 sectors)

1. Boot a 13-sector or 16-sector diskette containing NESTAR DISK COPY.
2. Type: MAXFILES 1 <cr>
3. Type: BRUN NESTAR DISK COPY <cr>
4. for DOS OR PASCAL(D/P)? enter: D

5. reply to remaining questions

WARNING: Copied Apple slave diskettes will not boot if the volume number is changed. (2)

When copying DOS 3.2 slave diskettes you will normally receive DRIVE ERROR messages for Track 0, Sectors 9,A,B, and C. These errors are not considered unusual, and the system will proceed as if they were not present.

B) Copying non-standard (no VTOC) BASIC or Pascal data diskettes. (13 or 16 sector)

1. Do 1 through 5 above
2. When NESTAR DISK COPY prints the following message:

READ (or DRIVE) ERROR [11,00] VTOC

PROCEED WITH FULL COPY (Y/N)?

Press Y.

3. NESTAR DISK COPY does not know whether you have a 13 or 16 sector DOS booted. It will print the message:

(A) 13-SECTOR OR (B) 16-SECTOR?

Press A or B.

NESTAR DISK COPY will now attempt to copy all sectors from the original to the copy.

C) Copying Pascal diskettes

1. Boot a 16 sector BASICs diskette (DOS 3.3) containing NESTAR DISK COPY
2. Type: MAXFILES 1
3. Type: BRUN NESTAR DISK COPY
4. For DOS OR PASCAL (D/P)? type: P
5. Reply to the remaining questions

NESTAR DISK COPY will now attempt to copy all sectors.

C.4 Technical Notes

NESTAR DISK COPY occupies memory from \$803 to \$FFF. The

DISK COPY

remaining available memory is used to buffer data being read and written.

NESTAR DISK COPY may be moved to other disks using the following steps:

1. BLOAD NESTAR DISK COPY
2. BSAVE NESTAR DISK COPY,A\$803,L\$7FD,Dd,ss

To re-enter NESTAR DISK COPY from BASIC, you may:

CALL 2051 (i.e. \$803).

Appendix D

Reading the Station Number From a Program

D.1 Introduction

It may be useful to read the station number (1 to 127) from within a program. This can provide a means of writing one standard HELLO program which then MOUNTs, RUNs, etc. different programs or disks based on the station number. A similar strategy can be used in Pascal by embedding the station number reading and a series of special MOUNTs, etc. within a program called SYSTEM.STARTUP.

The current station number is held in memory on the Nestar bus interface card, and can be read from there. This area of memory is addressable only by carefully following the Apple conventions for bank switching.

The following three program excerpts demonstrate how to access the one byte address field from Assembler, Basic or Pascal.

All three examples assume the Nestar bus interface card is in slot 6. If some other slot is used, the appropriate address (\$Cn00, where n is the slot number) should be used in the second step of the programs shown.

D.2 Machine Language

```
GETADR  LDA $CFFF          ; DESELECT ALL CARDS
        LDA $C600          ; SELECT BUS CARD (ASSUMING SLOT 6)
        LDA $C800          ; GET THE ADDRESS
        RTS                ; RETURN WITH IT IN A-REGISTER
```

D.3 (AppleSoft) BASIC

```
10 AD = PEEK(-12289) :REM DESELECT ALL CARDS
20 AD = PEEK(-14848) :REM SELECT BUS CARD (ASSUMING SLOT 6)
30 AD = PEEK(-14336) :REM NOW YOU HAVE THE ADDRESS 0 TO 127
```

READ STN #

D.4 Pascal

```
FUNCTION GETADDR:INTEGER;
```

```
TYPE BYTE = PACKED ARRAY[0..0] OF 0..255;
```

```
VAR TRIK: RECORD CASE BOOLEAN OF
```

```
    FALSE: (ADR:INTEGER);
```

```
    TRUE: (VAL:~BYTE)
```

```
    END;
```

```
BEGIN
```

```
    TRIK.ADR := -12289;
```

```
    GETADDR := TRIK.VAL^[0]; (* Deselect ALL CARDS *)
```

```
    TRIK.ADR := -14848;
```

```
    GETADDR := TRIK.VAL^[0]; (* SELECT CARD IN SLOT 6 *)
```

```
    TRIK.ADR := -14336;
```

```
    GETADDR := TRIK.VAL^[0]; (* RETURN ADDRESS VALUE *)
```

```
END;
```

Appendix E

Quick Reference List - Cluster Commands

	See also <u>Page</u>
BLOAD pathname [,LOAD=number]	5-5
-loads binary virtual diskette previously saved with BSAVE	
-loads at LOAD address in BLOAD, if present; if not, loads at LOAD address in corresponding BSAVE, if present; otherwise, loads at FROM address in BSAVE	
BRUN pathname [,LOAD=number]	5-5
-BLOADS a binary virtual diskette, then executes it starting at first address loaded	
BSAVE pathname, FROM=number, size [,LOAD=number] [,REPLACE]	5-5
-saves core image of specified size starting at FROM address on binary virtual diskette	
CREATE pathname, type [,size] [,sectors] [,protection]	5-9
-creates a new virtual diskette	
DELETE pathname	5-13
-deletes a virtual diskette from system	
HELP [command name] or ? [command name]	5-14
-displays list of commands or syntax of specified command	
LIST [pathname] [,VERBOSE] [,NESTED]	5-15
-lists entries in default or specified directory	
LOCK lockname [,usage]	5-18
-locks "lockname" for exclusive (EXC) or shared (SHR) use	
-Default: EXC	

COMMAND LIST

	See also <u>Page</u>
MOUNT pathname, drive [,usage]	5-20
-establishes correspondence between a virtual diskette and drive number d	
OFF	5-22
-initializes user station and runs /MAIN/BOOT	
PROTECT [pathname] [protection]	5-24
-modifies passwords or access rights for a virtual diskette or directory	
SET DIR [pathname]	5-27
-sets default directory	
SET [GRPPW=password] [PRVPW=password]	5-29
-sets default group or private password	
SHOW DATE	5-31
-displays date as MM/DD/YY	
SHOW DIR	5-32
-displays name of current default directory	
SHOW LOCK [lockname]	5-33
-lists all stations using specified lockname	
SHOW LOCKS	5-34
-displays all locks set at own station	
SHOW MOUNTS	5-35
-gives information on all virtual diskettes mounted at own station	

COMMAND LIST

See
also
Page

SHOW PROTECTION [pathname]	5-37
-displays group, private and public access rights for virtual diskette or directory addressed by pathname	
SHOW STATION	5-38
-displays own station number	
SHOW TYPE Dd	5-39
-displays type of virtual diskette mounted on drive d	
SHOW VOLS	5-40
-displays names of system disk units currently operational	
UNLOCK lockname or UNLOCK ALL	5-41
-releases locked usage on a lockname or on all locknames currently held	
UNMOUNT Dd or UNMOUNT ALL	5-42
-cancels previous mount(s)	

Parameters

access rights: set of letters from RWECD (read, write, erase, create, and delete)	4-2
drive: Dd, where d=number (1-255 for DOS; 4,5,9-12 for Pascal)	6-1 7-2
filepath: name[:password] [/filepath]	3-3
lockname: name of lock (same restrictions as name)	4-8
name: name of disk unit, directory, or virtual disk. (1-15 characters; no commas, slashes, colons, returns or unprintable characters; imbedded blanks ok; first character any ok character)	3-3
number: if decimal: between -32768 and 32767, inclusive if hex: between \$0 and \$FFFF, inclusive	3-3
password: a secret word used to gain access to protected Cluster-files. Same restrictions as name.	4-3

COMMAND LIST

	See also <u>Page</u>
pathname: /unitname/filepath or filepath (80 characters max. An initial slash indicates first field is a unit name; no initial slash indicates current default directory is to be prefixed to pathname.	3-3
protection: protect item [,protection]	4-3
protect item: ,GRPPW=password ,PRVPW=password ,PUBACC=access rights (Default: R) ,GRPACC=access rights (Default: PUBACC) ,PRVACC=access rights (Default: See 4.1.4)	4-3
sectors: SECTORS=13 for DOS 3.2.1; SECTORS=16 for DOS 3.3	5-9
size: SIZE=number [.size-unit] (can't be given for DOS, must be given for binary)	5-9
size-unit: C: character (1 byte)	5-9
S: sector (256 bytes)	5-5
B: block (512 bytes)	
blank: block (512 bytes)	
K: 1024 bytes	
type: T=B (binary) T=D (DOS) T=P (Pascal) T=Y (directory)	5-9
unitname: 1, 2, or 5 [:password] or name [:password]	3-1
usage: EXC (exclusive) or SHR (shared)	5-18
RO (read only) or RW (read/write)	5-20

Local CMD Commands

SET D<drive>,drive-status	7-4
SET D<drive>,S<slot>	7-5
SET S<slot>	7-5
SHOW DRIVES	7-4
SHOW SLOT	7-5
drive: 4, 9 or 11	7-2
drive-status: R, REAL, V, or VIRTUAL	7-2
slot: default virtual slot	7-5

A

access rights: 4-1, 4-4, 4-7
 ALL: 5-41, 5-42
 Apple: 1-1, 1-4
 Applesoft: 2-10, 6-3

B

barebones system: 2-1
 BASIC: 2-1..2-14, 6-1..6-5
 binary: 5-3, 5-5..5-8
 BLOAD: 5-5..5-8
 BOOT Program: 6-1, 7-2, B-1..B-3
 BRUN: 5-5..5-8, 5-22
 BSAVE: 5-5..5-8

C

Cluster: 1-1
 Cluster command: 5-1, 5-4, 6-2, E-1..E-4
 CMD: 7-3..7-4, E-4
 copying from a cassette tape: 2-13
 copying from a mini-diskette: 2-13, C-1..C-4
 copyright: 1
 crash: 7-3
 CREATE: 5-9..5-12
 ctrl-d: 2-10, 6-3, 6-4
 ctrl-n: 2-11, 6-3, 6-4
 ctrl-shift-n: 7-3

D

DATE: 5-31
 decimal numbers: E-3
 default access rights: 4-4..4-5
 default directory: 3-4, 5-3, 5-27..5-28, 5-32
 default parameters: 5-2
 default passwords: 4-6, 5-3, 5-29..5-30
 DELETE: 5-13
 directory: 2-2, 3-2
 default: 3-4, 5-3, 5-27..5-28, 5-32
 root: 2-2, 3-1..3-2
 subdirectory: 3-1..3-4
 disclaimer: 1
 disk controller: 1-4, 7-2
 disk copy routine: 2-13, C-1..C-4
 disk organization: 2-2, 3-1
 disk server: 1-1, 1-4
 disk units: 3-1, 5-40
 diskette: 2-1
 DOS with the Cluster/One system: 6-1

INDEX

drive: 5-20, 5-39, 5-42, 6-1, 7-1..7-3, 7-4, 7-5

E

error and information messages: A-1..A-4

EXC: 5-18, 5-20

exclusive usage: 5-18, 5-20

external devices: 1-3

F

field: 5-2

File Server: 1-1, 1-4

filename: 3-5

filepath: 3-5

FILER: 7-8

floppy disk: 1-2, 3-1

FROM: 5-5

G

group access rights: 4-2..4-5

GRPACC: 4-4..4-5, 5-10, 5-24..5-26

GRPPW: 4-4..4-5, 5-10, 5-24..5-26

H

hard disk: 1-2, 1-3

HELP: 5-13

hexadecimal number: E-3

How to Use This Manual: ii

I

IN#n: 2-11, 6-3, 6-4

INIT: 2-9

initializing a DOS virtual diskette: 2-9, 6-5

initializing a Pascal virtual diskette: 7-8

Introduction: 1-1

issuing Cluster commands within a BASIC program: 2-10..2-12, 6-2

issuing direct Cluster commands under DOS: 2-4..2-8, 6-2

J

K

keyword: 5-2

L

LIST: 5-15..5-17

LOAD: 5-5

LOCK: 4-8, 5-18..5-19, 5-33..5-34, 5-41
 lockname: 4-8, 5-18..5-19, 5-33..5-34, 5-41
 locks versus passwords: 4-1
 lowercase: 3-6

M

/MAIN: 2-3, 5-16..5-17
 MINICMD: 7-7..7-8
 minimum hardware requirements: 1-4
 MOUNT: 2-2, 4-6, 5-20..5-21, 5-35..5-36

N

NESTED: 5-15
 NETTALK: 7-5..7-8
 NETUNIT: 7-5..7-8

O

OFF: 5-22
 optional parameter: 5-2
 overview: 1-1

P

parameters: 5-2, E-3
 Pascal: 7-1..7-9
 Pascal with the Cluster/One System: 7-1..7-9
 password: 3-5, 3-6, 4-3
 pathname: 3-3..3-6, 5-23
 POKE: 6-1..6-2
 PR#n: 2-5, 2-10, 2-11, 6-3, 6-4
 preset passwords: 4-6
 private access rights: 4-2..4-5
 PROTECT: 4-5, 5-24..5-25
 protection: 4-1..4-9, 5-9..5-10, 5-24..5-26, 5-37
 PRVACC: 4-4..4-5, 5-10, 5-24..5-26
 PRVPW: 4-4..4-5, 5-10, 5-24..5-26
 PUBACC: 4-4..4-5, 5-10, 5-24..5-26
 public access rights: 4-2..4-5

Q

quick list of commands and parameters: E-1..E-4

R

reader comment form: ii
 REP: 5-5..5-6
 REPLACE: 5-5..5-6
 root directory: 2-2, 3-1..3-2

INDEX

RO: 5-20
RW: 5-20
RWECD: 4-1, 4-4, 4-7

S

sector: 5-9, 6-1
SECTORS: 5-9, 6-1
SENDMSG: 7-5..7-8
SET D<drive>,S<slot>: 7-5
SET D<drive>,drive-status: 7-4
SET DIR: 5-27..5-28
SET GRPPW: 5-29..5-30
SET password: 5-29..5-30
SET PRVPW: 5-29..5-30
SET S<slot>: 7-5
shared usage: 5-18, 5-20
SHOW DATE: 5-31
SHOW DIR: 5-32
SHOW DRIVES: 7-4
SHOW LOCK: 4-9, 5-33
SHOW LOCKS: 4-9, 5-34
SHOW MOUNTS: 5-35..5-36
SHOW PROTECTION: 4-6, 5-37
SHOW SLOT: 7-5
SHOW STATION: 5-38
SHOW TYPE: 5-39
SHOW VOLS: 5-40
SHR: 5-18, 5-20
simplest disk organization: 2-2
SIZE: 1-2, 5-5..5-6, 5-9, 6-1, 7-1
slot: 1-4, 6-2, 7-4, 7-5
space: 1-2
STATION: 5-38
station number: D-1..D-2
subdirectories: 3-1..3-4
syntax: 5-1
SYSTEM.APPLE: B-3
SYSTEM.MISCINFO: 7-8, B-3
SYSTEM.PASCAL: 7-8, B-3
SYSTEM.STARTUP: 7-8
system manager's manual: 3-2

T

tutorial for a bare-bones system: 2-1..2-14
type: 5-9, 5-39, 5-15..5-17

U

unit: 3-1, 5-40
unitname: 3-1, 3-3..3-4

UNLOCK: 4-9, 5-41
 UNMOUNT: 5-42
 usage: 5-18, 5-20
 user-classes: 4-2..4-3
 user-stations: 1-4
 utilities: 2-2, 3-2

V

VERBOSE: 5-15
 virtual diskettes: 2-8, 3-2
 virtual diskettes of variable size: 7-1
 virtual drives: 2-2
 volume: 3-1, 5-35

W

Warning: 5-42, 6-2
 warranty: i

X

Y

Z

@

@: 2-5, 5-1, 6-2

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306

READERS COMMENT FORM

CLUSTER/ONE MODEL A NESTAR SYSTEMS, INCORPORATED

You are encouraged to use this form to communicate to Nestar any problems, bugs or suggestions associated with Nestar products. We would like your comments on improving the system itself, as well as this documentation.

Nestar installation location:

Submitter's name :

Address :

Nestar environment

Model A or Model One:

Hard disk size (if present):

Number of user stations:

Language used:

Type of application:

Description of problem or suggestion:

Please mail this form to:

Nestar Systems, Incorporated
430 Sherman Avenue
Palo Alto, California 94306